

**UNIVERSIDADE FEDERAL DE SANTA CATARINA
DEPARTAMENTO DE INFORMÁTICA E ESTATÍSTICA
PROGRAMA DE PÓS-GRADUAÇÃO EM CIÊNCIAS DA
COMPUTAÇÃO**

Cecilia Estela Giuffra Palomino

**Modelo de Sistema Tutorial Inteligente para
Ambientes Virtuais de Aprendizagem baseado em Agentes**

Dissertação submetida ao Programa de
Pós Graduação de Ciências da
Computação para a obtenção do grau
de Mestre em Ciências da
Computação.

Orientador: Prof. Dr. Ricardo
Azambuja Silveira

Florianópolis
2013

Catálogo na fonte elaborada pela biblioteca da
Universidade Federal de Santa Catarina

Palomino, Cecilia Estela Giuffra
Modelo de Sistema Tutorial Inteligente para Ambientes
Virtuais de Aprendizagem baseado em Agentes [dissertação] /
Cecilia Estela Giuffra Palomino; orientador, Ricardo
Azambuja Silveira - Florianópolis, SC, 2013.
92 p.; 21cm.

Dissertação (mestrado) - Universidade Federal de Santa
Catarina, Centro Tecnológico. Programa de Pós-Graduação em
Ciência da Computação.

Inclui referências

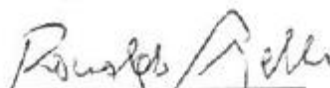
1. Ciência da Computação. 2. Sistema de Tutores
Inteligentes. 3. Ambientes Virtuais de Aprendizagem. 4.
Sistemas Multi-agente. I. Azambuja Silveira, Ricardo. II.
Universidade Federal de Santa Catarina. Programa de Pós-
Graduação em Ciência da Computação. III. Título.

Cecilia Estela Giuffrè Palomino

Modelo de Sistema Tutorial Inteligente para Ambientes Virtuais de Aprendizagem baseado em Agentes

Esta Dissertação foi julgada adequada para obtenção do Título de Mestre, e aprovada em sua forma final pelo Programa de Pós Graduação de Ciências da Computação

Florianópolis, 01 de fevereiro de 2013.

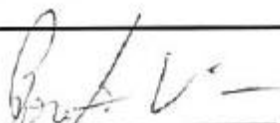


Prof. Ronaldo dos Santos Mello, Dr.
Coordenador do Curso

Banca Examinadora:



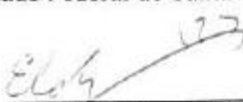
Prof. Dr. Ricardo Azambuja Silveira
Orientador
Universidade Federal de Santa Catarina



Prof.ª Dr.ª Rosa Maria Vicari,
Universidade Federal de Rio Grande do Sul



Prof.ª Dr.ª Silvia Modesto Nassar,
Universidade Federal de Santa Catarina



Prof. Dr. Elder Rizzon Santos,
Universidade Federal de Santa Catarina

À minha família, especialmente ao
meu avô e minha avó.

AGRADECIMENTOS

Ao meu pai, meu maior incentivador, pelos conselhos, pelas constantes comunicações online, pelas vezes que ficou feliz por todas as coisas boas que aconteciam comigo, pelo exemplo. Ao meu avô e minha avó que me apoiaram em todo momento, mesmo sabendo que estaria longe. A saudade que deixaram é grande. À minha mãe, meus irmãos, meus tios, primos, que de alguma forma ou de outra, estiveram presentes apoiando cada passo. À minha família em geral.

Ao meu orientador, professor Ricardo, pela confiança que depositou em mim e pela oportunidade que me deu de pertencer ao laboratório IATE, onde aprendi muito e pude me desenvolver como profissional. Pela dedicação, conversas, orientação, apoio. Pela preocupação constante, pela amizade.

Ao Glauco, meu namorado, meu companheiro, pelo apoio nesses anos todos, pela paciência nos momentos de desespero, pelas palavras de conforto nos momentos de fraqueza, por estar aí, por toda essa ajuda sem fim.

Aos meus amigos em geral, que sempre estiveram dispostos a ajudar. Aos meus amigos e colegas do IATE, em especial à Carla e Giane pelo apoio constante de diversas maneiras.

À coordenação do PPGCC, pela ajuda em todos os procedimentos necessários para chegar até aqui. Aos professores do programa.

À Universidade Federal de Santa Catarina, pela oportunidade.

Noventa por cento do sucesso se baseia
simplesmente em insistir.

(Woody Allen)

RESUMO

A educação assistida por computador é cada vez mais explorada, assim como o uso de ambientes virtuais de aprendizagem. Ambientes virtuais de aprendizagem (AVA) são utilizados tanto no ensino a distância quanto no presencial como ferramenta de apoio aos professores e alunos no processo de ensino-aprendizagem.

Os professores conseguem disponibilizar o material, fazer atividades e criar avaliações para os alunos. Porém, esse processo é feito da mesma forma para todos os alunos, sem levar em conta as diferenças entre eles, tanto em desempenho quanto em comportamento no ambiente.

A proposta deste trabalho é elaborar um modelo de ambiente inteligente de aprendizagem, inspirado no funcionamento de tutores inteligentes, baseado em agentes, para dar adaptabilidade a ambientes virtuais de aprendizagem distribuídos. Utilizando o AVA Moodle como estudo de caso, levando em conta o desempenho do aluno nas tarefas e atividades propostas pelo professor e acompanhando o acesso dele ao material de estudo.

Foi feito um estudo da base de dados do AVA utilizado no estudo de caso, realizada a extensão dessa base de dados, através da criação de novas tabelas e foi também modificado o código da instalação do AVA utilizado a fim de permitir a configuração dos agentes pelos professores, utilizando a interface do AVA. E foram, por fim, implementados os agentes para, através do envio de mensagens aos alunos e da configuração dos recursos e atividades a serem disponibilizados na disciplina, procederem as ações pedagógicas definidas pelo professor, de forma individualizada e adaptativa a cada aluno, em cada disciplina.

Palavras-chave: Ambientes Virtuais de Aprendizagem, Moodle, Sistemas de Tutores Inteligentes, Sistemas Multiagente.

ABSTRACT

The computer-assisted education is increasingly exploited, as well as the use of Learning Management Systems. Learning Management Systems (LMS) are used in distance learning and classroom teaching as teachers and students support tools in the teaching–learning process.

Teachers can provide material, do activities and create assessments for students. However, this process is done in the same way for all the students, regardless of their differences in performance and behavior in the environment.

The purpose of this work is to develop an agent-based intelligent learning environment model inspired by intelligent tutoring to provide adaptability to distributed LMS, using Moodle as a case study and taking into account students' performance on tasks and activities proposed by the teacher, as well as monitoring his/her study material access.

It was made a study of the LMS database used for the case study, was made an extension of this database creating new tables, and was modified the source code of the LMS installation in order to allow the agents configuration by the teachers, using the LMS interface. Finally were implemented the agents for, through the send of the messages to the students and the configuration of resources and activities to be made available in the discipline, respectively, proceed with the pedagogical actions defined by the teacher, in an individualized and adaptive way for each student in each discipline.

Keywords: Learning Management System, Moodle, Intelligent Tutoring System, Multi-Agent System.

LISTA DE FIGURAS

Figura 1. Modelo clássico com modelo proposto.....	48
Figura 2. Modelo de AVA em produção	50
Figura 3. Modelo do sistema	51
Figura 4. Diagrama de sequência	59
Figura 5. Seleção do recurso e atividade iniciais	67
Figura 6. Seleção de pré-requisitos	68
Figura 7. Grafo de dependências	70
Figura 8. Visão dos estudantes	79
Figura 9. Visão do professor	80
Figura 10. Ambiente inteligente de aprendizagem.....	82

LISTA DE TABELAS

Tabela 1. Mensagens de incentivo do Agente Tutor	73
Tabela 2. Mensagens de parabenização do Agente Tutor	74

LISTA DE ABREVIATURAS E SIGLAS

<i>ACISSE</i>	<i>Advances in Computer, Information, and Systems Sciences, and Engineering</i>
<i>ANGEL</i>	<i>A New Global Environment for Learning</i>
<i>AVA</i>	<i>Ambiente Virtual de Aprendizagem</i>
<i>AVEA</i>	<i>Ambiente Virtual de Ensino Aprendizagem</i>
<i>BDI</i>	<i>Belief-Desire-Intention</i>
<i>CARTAGO</i>	<i>Common ARTifact infrastructure for AGents Open environments</i>
<i>CMS</i>	<i>Course Management System</i>
<i>IA</i>	<i>Inteligência Artificial</i>
<i>ICALT</i>	<i>International Conference on Advanced Learning Technologies</i>
<i>IDE</i>	<i>Integrated Development Environment</i>
<i>ISKM</i>	<i>Intelligent Systems for Knowledge Management</i>
<i>ITS</i>	<i>Intelligent Tutoring System</i>
<i>KEWL</i>	<i>Knowledge Environment For Web-Based Learning</i>
<i>LMS</i>	<i>Learning Management System</i>
<i>MOODLE</i>	<i>Modular Object-Oriented Dynamic Learning Environment</i>
<i>OLAT</i>	<i>Online Learning and Training</i>
<i>PHP</i>	<i>Hypertext Preprocessor</i>
<i>SMA</i>	<i>Sistema Multiagente</i>
<i>SQL</i>	<i>Structured Query Language</i>

SUMÁRIO

RESUMO	31
ABSTRACT	33
LISTA DE FIGURAS	35
LISTA DE ABREVIATURAS E SIGLAS	39
1. INTRODUÇÃO.....	23
1.1 OBJETIVOS.....	24
1.1.1. Geral.....	24
1.1.2. Específicos.....	24
1.2. ESTRUTURA DO TEXTO.....	25
2. FUNDAMENTAÇÃO TEÓRICA	27
2.1. AMBIENTES VIRTUAIS DE APRENDIZAGEM.....	27
2.2. AGENTES INTELIGENTES.....	30
2.2.1. Agentes B.D.I.	33
2.2.2. Agentes Pedagógicos	34
2.3. SISTEMAS MULTIAGENTE	35
2.4. AGENTES E ARTEFATOS	37
2.5. SISTEMA DE TUTORES INTELIGENTES.....	38
2.6. AMBIENTES INTELIGENTES DE APRENDIZAGEM	39
3. TRABALHOS CORRELATOS	41
4. METODOLOGIA.....	45
5. DEFINIÇÃO DO MODELO	47
5.1. MODELO GERAL DO SISTEMA.....	50
5.1.1. Modelo do sistema tutorial	51
5.1.2. Modelo do aluno	54
5.2. COMO FUNCIONA O MODELO PROPOSTO?	55

6.	IMPLEMENTAÇÃO DO MODELO	61
6.1.	MOODLE.....	62
6.1.1.	Banco de dados do Moodle	64
6.2.	AMBIENTE DE DESENVOLVIMENTO JASON	64
6.3.	ETAPAS DE DESENVOLVIMENTO:.....	65
6.3.1.	Desenvolvimento do bloco tutor.....	66
6.3.2.	Desenvolvimento do cálculo de perfil do aluno (Bedel)	67
6.3.3.	Desenvolvimento do código de disponibilização dos recursos e atividades (Moodle)	71
6.3.4.	Desenvolvimento do Agente Tutor – mensagens de feedback para o aluno.....	72
7.	PROTÓTIPO DO MODELO.....	75
8.	CONSIDERAÇÕES FINAIS E TRABALHOS FUTUROS	81
	REFERÊNCIAS.....	83
	APÊNDICE A – Telas do sistema de configuração do agente.....	89

1. INTRODUÇÃO

A quantidade de alunos com acesso a um computador aumentou de maneira considerável nos últimos anos. Além disso, o ensino através do computador pode ser tão efetivo quanto o presencial. Ambos têm vantagens e desvantagens e, combinando ensino presencial com virtual, a melhora no aprendizado é mais significativa ainda (MORAN, 1994). O fato dos alunos procurarem a informação através do computador os converte em alunos mais ativos. “Existem atividades que podem ser realizadas com o computador que forcem o aluno a buscar informações, processá-las e utilizá-las na resolução de problemas, permitindo a compreensão do que faz e a construção do seu próprio conhecimento.” (VALENTE, 2009)

Ambientes virtuais de aprendizagem são definidos como ferramentas interativas para aprendizagem, onde o conteúdo está disponível *online*. Elas permitem que o professor forneça *feedback* aos alunos nas atividades de aprendizagem e são considerados recursos importantes para a educação de ensino superior. (AL-AJLAN, 2008)

Ambientes virtuais de aprendizagem são usados de forma satisfatória em *e-learning*, porém, via de regra, não atuam de forma interativa e personalizada com os alunos na disponibilização de tarefas e material de estudo, e fornecem os mesmos recursos de mediação pedagógica e o mesmo conteúdo para todos eles, sem considerar suas necessidades específicas. Atualmente, são feitas muitas pesquisas para incorporar recursos que levem em conta as características individuais dos estudantes, tais como estilos de aprendizagem, na tecnologia avançada de aprendizagem. De acordo com teorias educacionais, alunos com uma preferência forte por um estilo de aprendizagem podem ter dificuldades em aprender se seu estilo não é considerado no ambiente educacional. (GRAF, 2008)

“A fim de prover aos ambientes de aprendizagem, a capacidade de adaptação, de acordo com as características do aluno, além de permitir

um alto grau de interatividade entre o ambiente e os usuários, as pesquisas apontam para o uso de recursos propiciados pela Inteligência Artificial.” (SILVEIRA, 1998)

De acordo com isto surge a motivação desta pesquisa que se reflete no problema de como potencializar o processo de ensino-aprendizagem em ambientes virtuais de aprendizagem utilizando técnicas de inteligência artificial a fim de tornar o ambiente adaptativo às características dos estudantes, de forma individualizada?

Este trabalho propõe, portanto, utilizar arquiteturas baseadas em sistemas de tutores inteligentes e artefatos, aplicadas aos modernos ambientes virtuais de aprendizagem *online*, para conseguir um ensino personalizado, explorando as habilidades dos alunos da melhor forma, fazendo com que o aprendizado seja melhor e mais efetivo.

1.1 OBJETIVOS

Os objetivos, geral e específicos, são mostrados a seguir.

1.1.1. Geral

Elaborar um modelo de ambiente inteligente de aprendizagem, baseado em agentes, para ambientes virtuais de aprendizagem distribuídos, com características de tutor inteligente, com a finalidade de potencializar o processo de ensino-aprendizagem.

1.1.2. Específicos

- Elaborar um modelo de tutor inteligente compatível com ambientes virtuais de aprendizagem.
- Desenvolver o modelo do aluno, do professor e do tutor de acordo com a arquitetura dos ambientes virtuais de aprendizagem.
- Avaliar o modelo proposto, implementando as adaptações de um ambiente de aprendizagem para a integração com o tutor inteligente.

1.2. ESTRUTURA DO TEXTO

O capítulo 2 apresenta a fundamentação teórica deste trabalho, incluindo a pesquisa sobre ambientes virtuais de aprendizagem, agentes B.D.I., agentes pedagógicos, sistemas multiagentes, sistemas de tutores inteligentes e ambientes inteligentes de aprendizagem.

O capítulo 3 apresenta os trabalhos correlatos, o capítulo 4 a metodologia utilizada no trabalho. O capítulo 5 a definição do modelo utilizado para a criação dos agentes. O capítulo 6 trata a implementação do modelo. O capítulo 7 descreve a validação do modelo e, finalmente, o capítulo 8 mostra as considerações finais e trabalhos futuros.

2. FUNDAMENTAÇÃO TEÓRICA

Nesse capítulo serão apresentados os conceitos de ambientes virtuais de aprendizagem e, particularmente, o ambiente Moodle, utilizado como estudo de caso na avaliação do modelo proposto. Serão apresentados também conceitos de agentes inteligentes, agentes B.D.I (*Belief-Desire-Intention*) e agentes pedagógicos, sistemas multiagente, agentes e artefatos, sistemas de tutores inteligentes e ambientes inteligentes de aprendizagem.

São utilizados os agentes BDI, pela capacidade que eles têm de se adaptar às mudanças do ambiente, que no caso do ambiente virtual de aprendizagem muda constantemente pelas interações dos alunos e do professor em cada disciplina, e pela sua característica das crenças, desejos e intenções, para a implementação dos agentes.

O sistema de agentes é pensado como um sistema multiagente, pois é necessário utilizar os conceitos deste sistema em um contexto com muitas disciplinas, alunos e professores, pois para cada disciplina é criado um agente e para cada aluno também. E é necessário ter uma comunicação entre o agente da disciplina e os agentes dos alunos matriculados nela.

2.1. AMBIENTES VIRTUAIS DE APRENDIZAGEM

Ambientes virtuais de aprendizagem (AVA) são ferramentas e recursos tecnológicos que “utilizam o ciberespaço para veicular conteúdo e permitir interação entre os atores do processo educativo.” (PEREIRA, 2007).

Para SANTOS (2003),

Um ambiente virtual é um espaço fecundo de significação onde seres humanos e objetos técnicos interagem, potencializando assim, a construção de conhecimentos. Todo ambiente virtual é um ambiente de aprendizagem? Se entendermos aprendizagem como um processo

sócio técnico onde os sujeitos interagem pela cultura sendo esta um espaço para construção de saberes e conhecimento, então, se pode afirmar que sim.

Tendo em vista este fato, alguns autores preferem utilizar o termo Ambientes Virtuais de Ensino Aprendizagem (AVEA) para referir-se aos ambientes virtuais construídos com a finalidade de disponibilizar diversas ferramentas de comunicação e interação. “As tarefas de planejamento, elaboração e implementação da mediação pedagógica competem ao movimento intencional, organizado e sistemático da equipe envolvida” (RONCARELLI, 2007a), portanto a incorporação do termo “ensino” traduz a ideia da intencionalidade de mediação pedagógica.

Segundo RONCARELLI (2007b), “um AVEA diferencia-se de um AVA, pois contempla quatro pilares: sistematização, organização, intencionalidade pedagógica e caráter formal/institucional”.

Além disso, também se tornou bastante conhecido na literatura especializada o termo em inglês *Learning Management System* (LMS). Nesta dissertação, optou-se pela utilização simplificada do termo Ambientes Virtuais de Aprendizagem (AVA) para referir-se a estes ambientes projetados com a intencionalidade da mediação pedagógica, pela sua maior popularidade e simplicidade.

Com o avanço da tecnologia o uso destes ambientes tem aumentado pela sua facilidade em proporcionar a interação entre aluno e professor e a facilidade de acesso desde qualquer lugar e a qualquer momento, tanto aos conteúdos veiculados quanto às ferramentas de atividades e mediação pedagógica oferecidas pelo ambiente. Ambientes virtuais de aprendizagem disponibilizam diversas ferramentas de interação, tais como fóruns e chats, além de possibilitar a disponibilização de materiais pelos professores, como conteúdo de aprendizagem.

Nestes ambientes o fato dos alunos estarem em diferentes lugares geográficos e o fato de acessar em diferentes horários não é uma preocupação para eles. No entanto, existe a preocupação das

metodologias utilizadas precisarem ser diferentes para um ensino à distância em comparação a um ensino presencial. (TIJIBOY; PEREIRA; WOICIECHOSKI, 2009)

Segundo BIANCONCINI (2011),

Ambientes digitais de aprendizagem são sistemas computacionais disponíveis na internet, destinados ao suporte de atividades mediadas pelas tecnologias de informação e comunicação. Eles permitem integrar múltiplas mídias, linguagens e recursos, apresentar informações de maneira organizada, desenvolver interações entre pessoas e objetos de conhecimento, elaborar e socializar produções, tendo em vista atingir determinados objetivos.

Para DILLENBOURG (2000), ambientes virtuais de aprendizagem não se restringem à educação a distância. A educação baseada na web é frequentemente associada à educação a distância, porém, na prática, também é muito usada para apoiar o ensino presencial. O autor comenta, também, que a diferença entre estes dois tipos de educação está, pouco a pouco, desaparecendo. Muitos alunos de cursos de educação a distância não moram longe da escola, mas têm restrições de tempo, muitas vezes eles trabalham. Além disso, existem cursos que combinam distância e presença, o que faz os ambientes de aprendizagem serem mais robustos.

Ambientes virtuais de aprendizagem, em princípio usados basicamente em educação a distância, atualmente servem também de apoio nos cursos presenciais, como uma ferramenta de apoio ao professor para disponibilizar materiais, revisar trabalhos (tarefas), fazer o acompanhamento dos alunos na disciplina (relatórios de atividades) e também, para avaliá-los. No caso dos alunos, o ambiente facilita a entrega de trabalhos, a obtenção de material e a visualização das notas.

No que se refere ao aspecto pedagógico, os ambientes virtuais de aprendizagem colaborativa são espaços compartilhados de convivência que dão suporte à construção, inserção e troca de informações pelos

participantes, visando a construção social do conhecimento. (ROMANÓ, 2002)

Entre os ambientes virtuais de aprendizagem comerciais disponíveis, podemos citar os seguintes exemplos: *The Blackboard Learning System*, eCollege, ANGEL LMS, KEWL e Desire2Learn. Entre os ambientes virtuais de aprendizagem *Open Source* mais conhecidos estão: Moodle, Sakai, Dokeos, Claroline, e OLAT.

Ambientes virtuais de aprendizagem podem ser potencializados com técnicas de inteligência artificial, utilizando agentes inteligentes cooperativos (trabalham em background) ou pedagógicos animados (interagem com o usuário). No caso dos cooperativos, é feita uma modelagem multiagente, onde cada agente tem uma função específica e se comunica com os outros agentes. Estes agentes não são visíveis para o usuário. No caso dos agentes pedagógicos, são utilizados recursos multimídia para criar um personagem animado que interage com o aluno (Jaques, 2005), tendo como resultado, ambientes inteligentes de aprendizagem.

2.2. AGENTES INTELIGENTES

Um agente é uma entidade cognitiva ou uma abstração de algum dispositivo ou ser que pode perceber seu ambiente através de sensores e pode agir sobre este ambiente através de atuadores. “Um agente humano tem olhos, orelhas e outros órgãos como sensores, e mãos, pernas, boca, e outras partes do corpo como atuadores. Um agente robô tem câmeras e localizadores infravermelhos como sensores e vários motores como atuadores. Um agente de software tem cadeias de bits codificados como percepções e ações.” (RUSSELL, 1995)

Para WOOLDRIDGE (2009), um agente é um sistema de computador, situado em algum ambiente, capaz de realizar ações autônomas a fim de atender os objetivos que são delegados a ele. Nesse contexto, autonomia é ter habilidade para decidir como atuar, com a finalidade de realizar os objetivos.

Segundo BRENNER (1998), um agente requer sempre certa quantidade de inteligência para realizar suas tarefas. Um agente sem inteligência pode ser qualquer programa de software tradicional, pois ele também realiza tarefas específicas. Só a inteligência permite que um agente realize tarefas em grande parte de forma autônoma, requerendo a participação do usuário só para decisões importantes.

Ainda segundo BRENNER (1998), um agente deve interagir com seu ambiente para alcançar seus objetivos. Deve ser capaz de coletar informação no seu ambiente, tomar decisões baseado nessa informação, e depois iniciar ações específicas baseadas nas decisões.

Para WOOLDRIDGE (2009), agentes inteligentes são aqueles que têm, pelo menos, as seguintes características: autonomia, reatividade, proatividade e habilidade social.

- Autonomia significa ser capaz de funcionar e tomar decisões de forma independente.
- Reatividade é quando os agentes podem perceber seu ambiente e reponder, em tempo hábil, às mudanças que ocorrem nele para satisfazer seus objetivos.
- Proatividade é quando os agentes são capazes de exibir um comportamento direcionado a metas, tomando a iniciativa para satisfazer seus objetivos.
- Habilidade social em agentes é quando estes podem interagir com outros agentes (e possivelmente humanos) para satisfazer seus objetivos.

BRENNER (2008) descreve como características importantes de um agente inteligente, além das anteriores (autonomia, proatividade, reatividade e habilidade social), as seguintes: raciocínio/aprendizagem, mobilidade e caráter.

- Raciocínio/aprendizagem: a inteligência de um agente está formada por três componentes principais: sua base de conhecimento interna, a capacidade de raciocínio baseada nos

conteúdos da base de conhecimento e a habilidade de aprender ou se adaptar às mudanças no ambiente (comportamento adaptativo).

- Mobilidade: Descreve a habilidade de um agente para navegar dentro de redes de comunicação eletrônica. Reduz o carregamento da rede, pois um agente móvel não precisa reunir a informação que requer para cumprir suas tarefas enviando mensagens através da rede. Ele pode, ao invés disso, ir até o computador ou até os agentes que têm a informação que precisa, o que ocasiona um carregamento simples de rede, e ele realiza todas as tarefas localmente no computador alvo.
- Carácter: Em ocasiões, é desejável que um agente demonstre um comportamento externo com a maior quantidade possível de traços humanos, como uma pessoa virtual, por exemplo. As características mais importantes em um agente são honestidade, confiabilidade e segurança.

Para PADGHAM (2004), um agente inteligente é uma peça de *software* situada, autônoma, reativa, proativa, flexível, robusta e social.

- Situada: Existe em um ambiente.
- Autônoma: É independente e não é controlado externamente.
- Reativa: Responde em tempo hábil às mudanças no seu ambiente.
- Proativa: Persegue objetivos persistentemente.
- Robusta: Se recupera das falhas.
- Social: Interage com outros agentes.

Além dessas características, existem outras que não são menos importantes, porém, não são necessárias para todas as aplicações de agentes.

No contexto desse trabalho, um agente é uma entidade autônoma, capaz de tomar decisões, responder em tempo hábil, perseguir objetivos, interagir com outros agentes e que tem raciocínio e caráter, além de ser

do tipo B.D.I, com crenças, desejos e intenções, de acordo com os conceitos apresentados a seguir.

2.2.1. Agentes B.D.I.

Um agente racional é aquele que faz “a coisa certa”. A coisa certa é aquela que fará do agente o mais bem sucedido. Porém, é difícil decidir como e quando avaliar o sucesso do agente, pois depende do agente e da tarefa dele. Por exemplo, para um agente que deve limpar o chão, é medida a quantidade de sujeira que foi tirada em oito horas de deslocamento. Ou, para ser mais sofisticado, pode-se também levar em conta a quantidade de energia utilizada e o ruído gerado. Além disso, pode-se medir a eficiência dele verificando quanto foi limpo na primeira hora, ou seja, que tão rápido é o agente. (RUSSELL, 1995)

Um agente racional é aquele que escolhe realizar ações segundo seus próprios interesses, dadas as crenças que ele tem do mundo. Por exemplo, se alguém tem o objetivo de se manter seco, e tem a crença de que está chovendo, então, é racional pegar um guarda chuva na hora de ir para casa. O modelo B.D.I. (*Belief, Desire, Intention*) reconhece a importância das crenças, desejos e intenções nas ações racionais. (WOOLDRIDGE, 2000)

O modelo B.D.I. representa uma arquitetura cognitiva baseada em estados mentais, e tem sua origem no modelo de raciocínio prático humano. Uma arquitetura baseada no modelo BDI representa seus processos internos através dos estados mentais crença, desejo e intenção, e define um mecanismo de controle que seleciona de maneira racional o curso das ações. (FAGUNDES, 2004)

Segundo BORDINI (2007), quando se fala sobre sistemas B.D.I., está se falando de programas de computação com analogias computacionais de crenças, desejos e intenções.

- Crença: É a informação que o agente tem sobre o mundo. Ela pode não estar certa, ou ser imprecisa.

- **Desejo:** São todas as situações que o agente gostaria que acontecessem, as coisas que ele gostaria de realizar. Porém, ter um desejo não implica que o agente vá realizá-lo, mas é um potencial influenciador das ações do agente.
- **Intenção:** São as coisas que o agente decidiu realizar. Objetivos que são delegados ao agente. Pensa-se no agente olhando suas opções e escolhendo entre elas. As opções selecionadas se convertem em intenções, as quais o agente se compromete a realizar.

Um aspecto importante da arquitetura B.D.I. é a noção de compromisso com as decisões anteriores. Um compromisso incorpora o equilíbrio entre a reatividade e o direcionamento a um objetivo de um sistema orientado a agentes. Em um ambiente em mudança constante, compromissos dão certo senso de estabilidade ao processo de raciocínio de um agente. (RAO, 1995).

Agentes B.D.I. são agentes racionais que utilizam as próprias crenças que têm do mundo para realizar seus desejos, tendo a intenção de realiza-los. A arquitetura B.D.I. costuma ser empregada para a construção dos chamados agentes pedagógicos, descritos a seguir.

2.2.2. Agentes Pedagógicos

Agentes pedagógicos são aqueles cujo objetivo é ajudar os alunos no processo de ensino aprendizagem. Segundo GIRAFFA (1999), incorporar agentes a um programa educacional é intensificar os aspectos pedagógicos desejáveis no ambiente. Agentes pedagógicos podem ser divididos em agentes dirigidos a objetivos e agentes dirigidos a utilidades e têm como propriedades principais a autonomia, a habilidade social, a proatividade e a persistência.

Os agentes dirigidos a utilidades são os que realizam tarefas auxiliares ligadas às atividades pedagógicas. Isto é, ajudam os estudantes a encontrar programas específicos, arquivos, diretórios. Eles executam tarefas para os estudantes ou para o tutor. Agentes dirigidos a objetivos possuem mobilidade na WEB e operam em diferentes contextos

(textuais, hipermídia). Podem desenvolver atividades em colaboração ou em competição com alunos, dependendo de seus próprios objetivos. Eles podem ser tutores, mentores ou assistentes.

Agentes pedagógicos podem ser animados, e são personagens que facilitam o aprendizado em ambientes de aprendizagem baseados em computadores. Estes agentes têm entendimento suficiente do contexto de aprendizagem e do assunto e são capazes de realizar papéis úteis em cenários de aprendizagem. (JOHNSON, 1999)

Para GIRAFFA (1998), existem muitas motivações para usar uma apresentação animada de agente para propósitos de ensino-aprendizagem:

- Adiciona poder expressivo às habilidades de apresentação do sistema.
- Ajuda os estudantes a realizarem tarefas procedurais demonstrando-as.
- Serve como um guia através dos elementos do cenário (simulações)
- Podem envolver estudantes sem distraí-los ou distanciá-los da experiência de aprendizagem.

Na prática, sistemas de agentes únicos são raros. O mais comum são os casos de agentes que habitam um ambiente que contém outros agentes.

2.3. SISTEMAS MULTIAGENTE

Para WOOLDRIDGE (2009),

Um sistema multiagente (SMA) é um sistema com vários agentes que se comunicam entre si, e que atuam em um ambiente. Diferentes agentes têm diferentes “esferas de influência” no sentido que eles terão controle sobre diferentes partes do ambiente. Em alguns casos estas esferas podem coincidir e, o fato delas coincidirem, pode aumentar a dependência entre os agentes.

Em um domínio multiagente o mais importante é saber qual será o tipo de interação que será realizada entre os agentes com o objetivo de serem capazes de tomar a melhor decisão possível sobre que ação realizar.

O enfoque principal dos sistemas multiagente é prover mecanismos para a criação de sistemas computacionais a partir de entidades de software autônomas, denominadas agentes, que interagem através de um ambiente compartilhado por todos os agentes de uma sociedade. (BORDINI, 2001).

Segundo BORDINI (2001), existem dois grandes tipos de sistemas multiagente: reativos e cognitivos. Os sistemas reativos seguem a ideia de que um comportamento inteligente em um sistema emerge da interação entre um grande número de agentes muito simples. Estes agentes agem sob um esquema estímulo-resposta. Os sistemas com agentes cognitivos possuem em geral poucos agentes, pois cada agente é um sistema complexo e computacionalmente pesado, englobando as seguintes características: percepção, ação, comunicação, representação, motivação, deliberação e raciocínio e aprendizagem.

- Percepção: O agente é capaz de perceber alterações no ambiente.
- Ação: As alterações no ambiente são provenientes de ações que os agentes realizam nele.
- Comunicação: Como os agentes precisam coordenar suas ações, a comunicação entre eles é essencial.
- Representação: O agente possui uma representação simbólica explícita daquilo que acredita ser verdade em relação ao ambiente e aos agentes que compartilham este ambiente.
- Motivação: O agente age sobre o ambiente por iniciativa própria, para satisfazer seus objetivos.
- Deliberação: O agente tem que ser capaz de decidir quais, de fato, serão os objetivos a serem seguidos por ele.
- Raciocínio e aprendizagem: Criação de mecanismos de aprendizagem específicos para ambientes multiagentes.

A área de SMA estuda o comportamento de um grupo organizado de agentes autônomos que cooperam na resolução de problemas que estão além das capacidades de resolução de cada um individualmente. Duas propriedades, aparentemente contraditórias, são fundamentais para os SMA: a autonomia dos agentes e sua organização (BRIOT; DEMAZEAU, 2002 apud HUBNER, 2003), que estabelece restrições aos comportamentos dos agentes, visando estabelecer um comportamento grupal coeso.

2.4. AGENTES E ARTEFATOS

Recentemente, as propostas de modelagem de SMA se baseiam em duas abstrações distintas: agentes e artefatos (A&A), onde o agente é um ente (pró-) ativo que se encarrega de controlar, e realizar os objetivos através de tarefas, e o artefato um ente reativo cujas funções e serviços fazem com que os agentes individuais realizem trabalhos em conjunto em um SMA (OMICINI; RICCI; VIROLI, 2008).

Segundo (RICCI; PIUNTI; VIROLI, 2011) dentro de um *workspace* as ações disponíveis para os agentes estão separadas em três grupos: ações para criar, eliminar e procurar artefatos, ações para usar artefatos, executar operações e observar propriedades e sinais e ações para vincular e desvincular artefatos.

Os artefatos devem ser criados pelos agentes em tempo de execução. Tres ações básicas existem para este propósito: `makeArtifact`, `disposeArtifact` e `lookupArtifac`. A primeira ação instancia um novo artefato, a segunda permite remover um artefato do *workspace* e a última recupera um identificador único do artefato dado o nome lógico deste. (RICCI; PIUNTI; VIROLI, 2011)

O A&A é um meta-modelo de programação de agentes que utiliza, por exemplo, o framework CArtaGo (Common ARTifact infrastructure for Agents Open environments) para a implementação do artefato. Os agentes podem utilizar os recursos e ferramentas do artefato, de forma dinâmica, servindo como apoio nas suas atividades individuais e coletivas. (OMICINI; RICCI; VIROLI, 2008)

CARtAgO é um framework de propósito geral que possibilita a programação e execução de ambientes virtuais para sistemas multiagentes baseando-se no meta-modelo A&A para modelar e projetar estes sistemas. CARtAgO é mais efetivo quando é integrado com linguagens de programação de agentes inteligentes, em particular aqueles baseados na arquitetura BDI. (CARTAGO, 2013)

2.5. SISTEMA DE TUTORES INTELIGENTES

Sistemas de tutores inteligentes (ITS - *Intelligent Tutoring System*) são sistemas complexos que envolvem vários tipos diferentes de especialidade: conhecimento do assunto, conhecimento do conhecimento do aluno, conhecimentos pedagógicos, entre outros. Um ITS deve ser considerado sempre em evolução constante (FRASSON, 1997). Segundo SANTOS (2001), um ITS se caracteriza por incorporar técnicas de IA no seu projeto de desenvolvimento e atua como auxiliar no processo de ensino-aprendizagem.

Segundo CONATI (2009), sistemas de tutores inteligentes é o campo interdisciplinar que investiga como elaborar sistemas educacionais que fornecem instruções adaptadas às necessidades dos estudantes, como muitos professores fazem.

Pesquisas em ITS têm investigado como fazer tutores baseados no computador mais flexíveis, autônomos e adaptativos às necessidades de cada estudante, dando-lhes conhecimento explícito dos componentes relevantes do processo de ensino e habilidades de raciocínio para converter este conhecimento em comportamento inteligente.

Para GIRAFFA (1998), desenvolvimentos em ITS consideram uma abordagem cooperativa entre o estudante e o sistema. Segundo Oliveira (2005) O objetivo dos ITS consiste em complementar ou substituir um tutor humano, com a vantagem de acompanhar o aluno em cada passo de aprendizado.

As pesquisas em sistemas de tutores inteligentes preocupam-se com a construção de ambientes que possibilitem um aprendizado mais

eficiente. A tecnologia dos agentes tornou os ITS mais adaptados às necessidades e características individuais de cada estudante. (FRIGO, 2004).

Segundo VICCARI (1989),

Um tutor inteligente necessita explorar os conteúdos, possuir vários planos de ensino e um modelo para guiar a apresentação do conteúdo, ser sensível às necessidades do utilizador adequando-se às necessidades individuais, dominar o máximo possível o assunto que ensina, possuir conhecimento para tentar resolver situações não previstas nas regras existentes e aprender com tais situações, possuir características de ensino assistido, possuir mecanismos para a depuração inteligente e a orientação na detecção e eliminação de falhas, permitir a simulação automática e conduzida de problemas, além de possuir memória retroativa que descreva o raciocínio utilizado pelo aluno e pelo tutor durante a exploração de determinado conteúdo.

Sistemas de tutores inteligentes oferecem flexibilidade na apresentação do material e maior habilidade para responder às necessidades do aluno. Procuram, além de ensinar, aprender informações relevantes sobre o estudante, proporcionando um aprendizado individualizado. Sistemas de tutores inteligentes têm sido apresentados como altamente eficientes para a melhora do desempenho e motivação dos estudantes. (LIMA, 2004).

Sistemas de tutores inteligentes em ambientes virtuais de aprendizagem potencializam o processo de ensino-aprendizagem, fazendo do ambiente virtual um ambiente inteligente de aprendizagem.

2.6. AMBIENTES INTELIGENTES DE APRENDIZAGEM

Segundo PEREIRA (2007), ambientes virtuais de aprendizagem são ferramentas e recursos tecnológicos que utilizam o ciberespaço para

veicular conteúdo e permitir interação entre os atores do processo educativo.

Ambientes inteligentes de aprendizagem, por sua vez, são ambientes virtuais que incluem uma situação de solução de problema e um ou mais agentes que ajudam ao estudante nesta tarefa e monitoram seu aprendizado. Ambientes inteligentes de aprendizagem são ambientes virtuais nos quais são aplicadas técnicas de inteligência artificial. (DILLENBOURG, 1993)

Um ambiente inteligente de aprendizagem é um tipo de sistema educacional inteligente que combina características do sistema de tutor inteligente tradicional e dos ambientes de aprendizagem. (BRUSILOVSKY, 1994)

Segundo SILVEIRA (1998), o ambiente inteligente de aprendizagem deve construir e atualizar o modelo de aluno em função do que ele já sabe, o que pode variar significativamente de um aluno para outro. Esta diferença deve ser devidamente considerada na busca da eficiência, quando se desenvolve ambientes de ensino inteligentes, e considerar variáveis como a experiência prévia, os propósitos do aluno ou a experiência prévia do aluno em aprendizagem e suas expectativas.

No capítulo seguinte será aprofundado o estudo dos ambientes inteligentes de aprendizagem, através da análise de trabalhos correlatos a esta proposta.

3. TRABALHOS CORRELATOS

Com o intuito de conhecer o estado atual das pesquisas realizadas sobre ambientes virtuais de aprendizagem e o uso de agentes inteligentes como tutores nesses ambientes, foi realizada uma revisão sistemática da literatura, disponível nas principais bases de dados acessíveis através da Internet, prospectando os principais veículos de divulgação científica na área.

As fontes utilizadas na pesquisa foram o Portal da Capes, o IEEEExplore, a ACM Digital Library e a Springer Link.

Foram escolhidos trabalhos e artigos desenvolvidos e publicados entre 2004 e 2013. Foram feitas 14 pesquisas com diferentes conjuntos de palavras chave e foram encontrados 46 resultados relevantes, escolhidos pelo título e resumo, nos quais havia 31 trabalhos diferentes. Dos 31 selecionados, 22 estão disponíveis para leitura.

Foram encontrados trabalhos relacionados à tutoria em ambientes virtuais de aprendizagem e ambientes adaptativos, tendo como referência de AVEA o Moodle. Utilizando, principalmente, modelos de agentes inteligentes.

Os artigos selecionados podem ser divididos em algumas áreas:

- Agentes tutores inteligentes
- Tutores e adaptabilidade
- Estilos de aprendizagem
- Grupos de alunos com o mesmo estilo de aprendizagem
- Sistemas de aprendizagem adaptativos
- Comparação de ambientes virtuais de aprendizagem com o Moodle
- Importância de plataformas de código aberto
- Modelo de comportamento afetivo para sistemas de tutores inteligentes
- Criação adaptativa de auto-avaliações

Entre eles, os artigos que mais se assemelham com a proposta deste trabalho são três:

Approach to an Adaptive and Intelligent Learning Environment. (publicado no ACISSE em 2006): Nesse trabalho é proposta uma abordagem orientada a agente para o projeto e implementação de um componente adaptativo e inteligente para um Ambiente Virtual de Aprendizagem. A adaptatividade no modelo é definida como uma habilidade do sistema para criar e, durante o processo de aprendizagem, atualizar uniformemente o currículo que satisfaz as necessidades do estudante. O componente é um plug-in na distribuição padrão do ambiente virtual Moodle. O modelo proposto tem três partes que descrevem as funcionalidades principais do componente adaptativo e inteligente: montagem do currículo do estudante, ensinar ao estudante, controlar ao estudante. Na primeira, o estudante escolhe os cursos baseado em suas necessidades, o nível de excelência que ele quer alcançar e sua preferência em relação ao tipo de material de estudo. Na segunda parte o sistema decide como vai atuar, por exemplo, mostrar o material ao aluno, baseando-se na crença (modelo do aluno) que tem sobre ele. Na última parte é decidido quando fazer um teste de avaliação para o estudante ou outra atividade que possa avaliar algum conhecimento específico dele em relação ao currículo. Após completar a avaliação, a crença sobre o estudante é atualizada. (BAZIUKAITÉ, 2006).

Cluster Analysis in Personalized E-Learning. (publicado no ISKM em 2009): É proposta uma arquitetura de sistema, na qual técnicas de ensino e *layouts* apropriados são ajustados para grupos de estudantes com preferências similares, criada aplicando técnicas de *clustering*. São examinadas diferentes técnicas de *clustering* conhecidas para obter grupos da melhor qualidade, que são definidos na fase de pré-processamento. Materiais didáticos e conteúdo podem ser adaptados às necessidades de cada grupo e diferentes caminhos de aprendizagem podem ser criados. Os estudantes novos preenchem um questionário para determinar seu estilo de aprendizagem junto com suas escolhas de usabilidade e segundo isso o grupo apropriado é escolhido para cada um

deles. Estudantes que não se ajustam a nenhum grupo são considerados indivíduos. A ideia da solução proposta é dividir o processo em duas etapas: a primeira para procurar grupos de estudantes com uma similaridade grande e detectar aqueles isolados. Na segunda etapa, os grupos são misturados em grupos maiores se for necessário e os isolados são indicados. O objetivo dos experimentos foi examinar o desempenho da técnica de *clustering* proposta para diferentes conjuntos de dados de alunos dependendo da escolha dos parâmetros. Nesse capítulo considerou a aplicação da análise *cluster* para grupos de estudantes de acordo com seus estilos de aprendizagem dominantes e preferências de usabilidade. Usar uma classificação sem supervisão permite criar grupos de estudantes dinamicamente, dependendo dos cursos que eles frequentam. (ZAKRZEWSKA, 2009).

Supporting Cognitive Competence Development in Virtual Classrooms. (publicado no ICALT em 2010): A abordagem descrita nesse artigo implementa um mecanismo para criar adaptativamente questionários de auto-avaliação em um ambiente Moodle. O LMS é capaz de guardar todas as atividades *online* dos alunos em arquivos *log*. Esta informação pode ser usada, também, para gerar automaticamente um *feedback* inteligente para o aluno. As perguntas são derivadas de uma ontologia de competências que também é usada para indexar os materiais de aprendizagem. Os rastros do aluno através do material de aprendizagem são usados para determinar o estado atual do “conhecimento esperado” ou competências. O sistema inclui dois agentes principais, o agente gerente de objetivos, que guia o aluno no planejamento das atividades e o agente gerente de conteúdo, que guia o aluno durante a revisão dos recursos. Nesse artigo, foi apresentada uma extensão do LMS Moodle na qual são usadas ontologias para estruturar o processo de aprendizagem, fornecendo recursos e gerando questionários automaticamente para auto-avaliação dos estudantes. (WEINBRENNER, 2010)

Na pesquisa realizada para a análise do estado da arte foram encontrados trabalhos que tratam da adaptabilidade em ambientes virtuais de aprendizagem levando em conta as necessidades do estudante, estilos de

aprendizagem e preferências de usabilidade e dados do relatório de atividades (log) deles.

O presente trabalho propõe unir os dados que podem ser obtidos no banco de dados (informações do desempenho do aluno e logs) que existem usualmente na maioria dos ambientes virtuais de aprendizagem mais utilizados atualmente, para poder configurar a disciplina de forma personalizada para cada aluno, tanto no que se refere ao material disponibilizado para o aluno quanto às atividades propostas a ele, explorando as suas habilidades e contornando as suas deficiências. Tendo sempre uma linha base com material e atividades obrigatórias, e com as atividades fora desta linha divididas em diferentes níveis de dificuldade.

4. METODOLOGIA

Este trabalho é considerado uma pesquisa aplicada, pois tem como objetivo gerar conhecimentos para aplicação prática, dirigidos à solução de problemas específicos. (SILVA, 2005)

É uma pesquisa exploratória, pois envolve um estudo de caso, integrando o ambiente virtual Moodle com tutores inteligentes para desenvolver um ambiente inteligente de aprendizagem. (GIL, 1991)

O desenvolvimento deste trabalho está dividido nas seguintes etapas:

1. Fundamentação teórica: É feita uma revisão bibliográfica, da área de agentes inteligentes e pedagógicos, e de ambientes virtuais de aprendizagem. Após esta etapa, se tem o embasamento teórico necessário para propor o desenvolvimento de um tutor inteligente para ambientes virtuais de aprendizagem, resultando em ambientes inteligentes de aprendizagem.
2. Revisão do estado da arte: É feita uma pesquisa sistemática, com a finalidade de conhecer os trabalhos desenvolvidos relacionados com o tema desta pesquisa, para conhecer o que já foi desenvolvido e o que ainda falta desenvolver em ambientes inteligentes de aprendizagem.
3. Elaboração dos modelos dos tutores: É definido o modelo de tutor inteligente que será aplicado no ambiente virtual de aprendizagem. Além disso é definido o modelo do aluno.
4. Implementação do modelo: É feito o desenvolvimento do modelo de agentes.
5. Avaliação/testes do modelo, através da adaptação de um ambiente de aprendizagem para a integração com o tutor inteligente: Como estudo de caso, é feita uma adaptação do ambiente virtual de aprendizagem Moodle, que serve de ambiente de interação com os agentes que implementam os tutores inteligentes.

5. DEFINIÇÃO DO MODELO

Neste capítulo é apresentado o modelo proposto, iniciando com uma visão geral, seguido do detalhe do modelo geral do sistema, incluindo a definição dos modelos de tutor e aluno e finalmente é explicado o funcionamento do modelo.

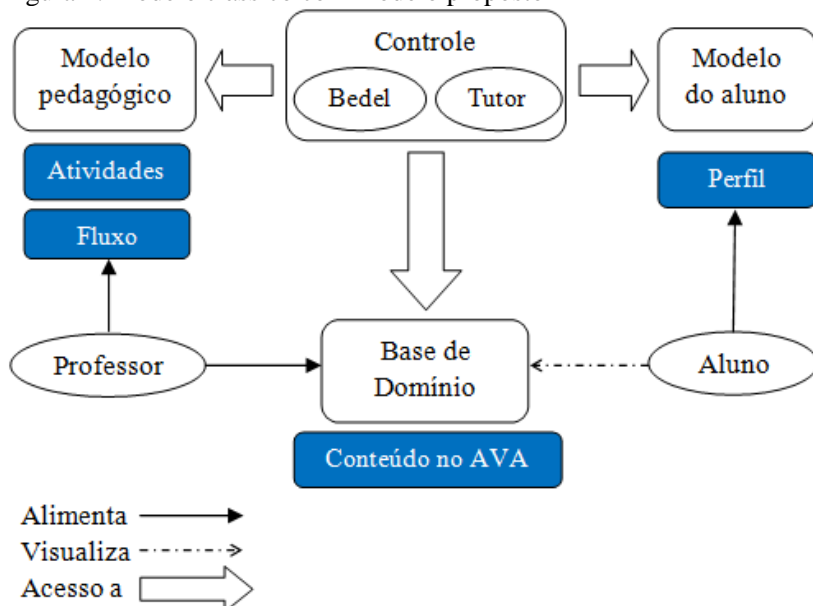
O presente trabalho consiste em criar uma arquitetura de agentes e uma base de conhecimento destes agentes que compõem o sistema tutorial inteligente com informações obtidas a partir do banco de dados de um ambiente virtual de ensino-aprendizagem. Para tanto se realiza um estudo de caso utilizando como base a arquitetura da plataforma Moodle, escolhida por ser uma plataforma muito utilizada atualmente, consolidada, do ponto de vista de funcionamento, e também por ser usada formalmente na instituição onde a pesquisa é realizada.

É definido o modelo dos agentes, que se conectam com o ambiente virtual de aprendizagem através do banco de dados. É desenvolvida a interface na qual o professor configura os níveis e prioridades dos recursos e tarefas no ambiente, é feita a adaptação do banco de dados com a criação das tabelas dos tipos de perfil, das notas perfis dos alunos (calculadas com as notas das atividades e os acessos aos recursos), das médias das notas perfis, da disponibilidade segundo o perfil, das tarefas segundo o nível, das tarefas avaliadas e das dependências dos recursos e atividades configurados pelo professor e é feita a integração das ações dos agentes com o ambiente virtual de aprendizagem Moodle.

O modelo clássico de sistemas de tutores inteligentes contém o modelo pedagógico, o modelo do aluno, a base de domínio, além do controle. No modelo proposto, dois tipos de agentes, denominados BEDEL e TUTOR são utilizados. Os agentes “bedel”, e toda a sua estrutura de conhecimento e interação, constituem o que corresponde, no modelo clássico dos tutores inteligentes, ao Modelo Pedagógico, enquanto que os agentes “tutor”, e toda a sua estrutura, constituem aquilo que, no modelo dos tutores inteligentes, corresponde ao Modelo do Aluno. O banco de dados do ambiente virtual de aprendizagem, por sua vez, pode

ser associado à abstração da base de domínio. Estas correlações são mostradas na Figura 1.

Figura 1. Modelo clássico com modelo proposto



No modelo proposto, o professor escolhe o modelo pedagógico a ser usado na disciplina, o material didático (leituras, artigos, vídeos) e atividades (trabalhos e provas) que ele considera necessários para explorar da melhor forma todos os recursos da disciplina, durante o semestre, e configura o Agente Bedel segundo este modelo através da interface do AVA. Este agente atua como um ajudante do professor. É o tutor da disciplina.

A elaboração do modelo pedagógico, a ser usado em cada disciplina, sob o controle dos agentes, é feita pelo professor, utilizando os recursos e atividades já disponíveis no ambiente virtual de aprendizagem, para disponibilizar o material e as tarefas para os alunos e utilizando a ferramenta especialmente desenvolvida, que se incorpora á interface do

AVA, a fim de definir um grafo que representa os diversos seqüenciamentos possíveis, para a mediação pedagógica, de acordo com o desempenho do aluno, na medida em que ele cursa a disciplina, realizando as atividades definidas no modelo elaborado pelo professor.

A escolha dos recursos e atividades, juntamente com a elaboração do fluxo, através da ferramenta desenvolvida, determina o comportamento dos agentes, pois eles são configurados levando em conta os recursos já existentes no ambiente virtual e seus planos e suas crenças são influenciados dinamicamente pelas alterações realizadas na base de dados do AVA, na medida em que transcorre o curso, e de forma individualizada para cada estudante.

O Agente Tutor é o agente que tem contato com o aluno. Ele orienta o aluno, indicando as mudanças no desempenho dele, cada vez que uma atividade é avaliada, incentivando-o a melhorar quando ele tiver tido uma queda no desempenho ou parabenizando-o quando ele tiver tido um desempenho melhor. Estas orientações são enviadas ao estudante através de mensagens do AVA.

O modelo proposto tem como objetivo criar uma base de conhecimento para os agentes, utilizando o banco de dados do AVA para obter e guardar as informações atualizadas por estes. Com estas informações o Agente Bedel atualiza os dados de perfil do aluno e informa o Agente Tutor das mudanças no perfil deste aluno.

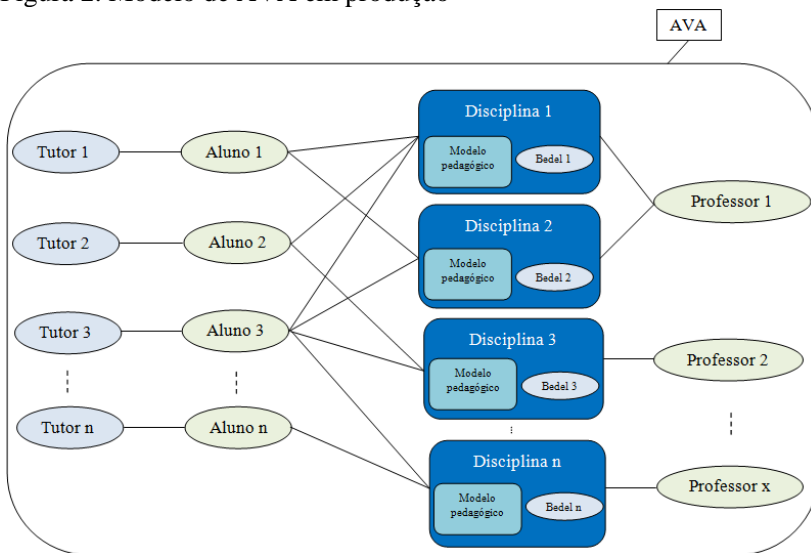
A proposta do modelo considera um ambiente virtual (Figura 2), com uma quantidade grande de alunos, professores e disciplinas, prevendo, por isso a existência de um Agente Bedel para cada disciplina e um Agente Tutor para cada aluno, sendo que este Agente Tutor verifica o desempenho do aluno em todas as disciplinas em que este está matriculado.

Dessa forma, o modelo propõe um sistema multiagente, com diversos agentes tutor se comunicando com o Agente Bedel de cada uma das

disciplinas e agentes bedel disponibilizando as atividades e recursos para todos os alunos da disciplina de forma personalizada.

Este cenário, dinâmico e complexo, demanda a necessidade de agentes autônomos e independentes que colaboram mutuamente para atingir os propósitos definidos, de forma global, de atender as diretrizes propostas por cada professor, em cada disciplina, na elaboração do modelo pedagógico e do modelo de domínio de cada uma delas.

Figura 2. Modelo de AVA em produção

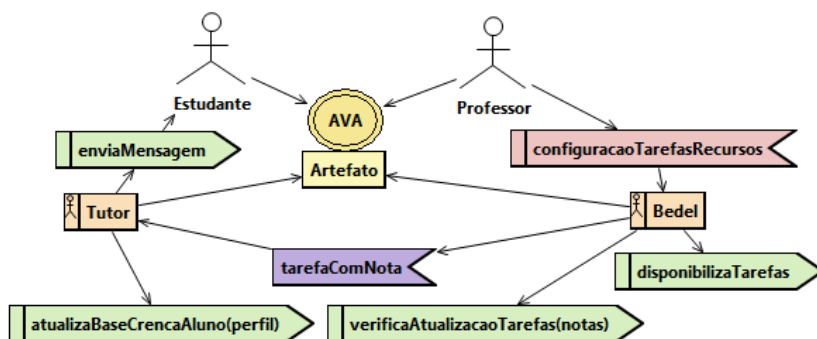


5.1. MODELO GERAL DO SISTEMA

Nesse trabalho são desenvolvidos os agentes segundo o modelo do tutor, explicado a seguir. Na Figura 3 pode-se ver o modelo do sistema, que contém o Ambiente Virtual de Aprendizagem, os atores, os agentes Tutor e Bedel e as diferentes interações entre eles. O ator “professor” é a figura do professor da disciplina que insere os recursos e atividades no ambiente e os configura para que o Agente Bedel saiba de que forma mostrá-los aos alunos.

Os atores estudante e professor são os usuários do ambiente virtual de aprendizagem que têm contato com a interface do AVA. O artefato é usado pelos agentes para a manipulação dos dados através do banco de dados do ambiente virtual. O Tutor e o Bedel são os agentes que realizam as diferentes ações. O Tutor atualiza a base de crenças que tem do aluno e envia mensagens para ele. O Bedel verifica a atualização das notas nas tarefas e disponibiliza as próximas segundo o perfil de cada um dos alunos. Além disso, envia uma mensagem ao Agente Tutor avisando que já teve uma tarefa avaliada (tarefaComNota) e as informações do perfil atualizado do aluno.

Figura 3. Modelo do sistema



5.1.1. Modelo do sistema tutorial

O modelo do sistema tutorial neste trabalho está formado por dois tipos de agentes com papel de tutor: o tutor da disciplina (denominado Agente Bedel) e o tutor que tem contato com o aluno (denominado Agente Tutor). O Bedel se encarrega de verificar o aproveitamento do aluno nas diferentes atividades. Este agente é criado para cada disciplina depois que o professor configura o agente. O Agente Tutor se encarrega de entrar em contato com aluno, através de mensagens, para incentivá-lo ou parabenizá-lo, segundo o desempenho dele. Este agente é criado para cada aluno.

Cada vez que o aluno acessa o ambiente virtual de aprendizagem, o Agente Tutor verifica a informação que tem deste aluno no banco de dados e atualiza as crenças que tem dele. Após isto, interage com o aluno utilizando a funcionalidade de mensagens do AVA.

O agente da disciplina (Bedel) verifica constantemente a avaliação das tarefas, por parte do professor, calcula o desempenho dos alunos, atualiza a crença que tem deles e guarda no banco de dados as informações necessárias para que no AVA sejam disponibilizadas para eles novas atividades, de forma personalizada, levando em conta as suas notas nas atividades e as interações nos diferentes recursos da disciplina.

As crenças iniciais do Agente Tutor são: *notaPerfilAluno* e *atividadeAvaliada*, inicializada com o valor “false”. O Agente Tutor recebe do Agente Bedel a informação de que uma atividade teve avaliação e atualiza a crença que tem sobre as atividades avaliadas e a sua crença do aluno, no que se refere à nota perfil dele.

O Agente Tutor executa os planos *atualizaNota* e *enviaMensagem*, onde atualiza a crença do aluno e envia mensagem para ele segundo a nova nota perfil recebida.

As crenças iniciais do Agente Bedel são *acessoProfessor*, *olhoFechado* e *ativou Banco*. Todas inicializadas com o valor “false”. Estas crenças estão relacionadas com a disciplina. Para cada disciplina existe um artefato com id igual ao id da disciplina. A crença *acessoProfessor* é atualizada no primeiro acesso do professor à disciplina. Verificando isto, o agente sabe que o professor entrou na disciplina e ativou o bloco de configuração do Bedel.

A crença *olhoFechado* é atualizada após o professor inserir os recursos e atividades da disciplina e configurar o Agente Bedel. Esta crença serve para verificar que todos os recursos e atividades estão ocultos para os alunos ao iniciar o estudo.

Os planos do Agente Bedel são executados da seguinte forma: O agente começa ativando o artefato da disciplina, que controla a conexão dos

agentes com o banco de dados do AVA, após ter sido criada a disciplina. Em seguida, o Bedel verifica o primeiro acesso do professor, verifica os recursos e atividades existentes na disciplina e, interagindo com o banco de dados, faz com que os mesmos fiquem ocultos para os alunos, com exceção do recurso e atividade iniciais.

Após isto, o agente ativa um contador que, a cada 24 horas, faz com que ele execute o plano *verifica_data_final_tarefa* no qual ele verifica se alguma tarefa passou da data de entrega e, em caso positivo, executa o plano *verifica_avaliação*, onde ele verifica se o professor avaliou a tarefa “vencida”. Caso contrário, espera até o próximo “tick” para executar novamente esse plano.

Na verificação da avaliação da tarefa pelo professor, ele confere que todos os alunos tenham sido avaliados e, após isto, calcula a nota perfil deles, a média das notas perfil e os valores de cada um dos perfis (básico, intermediário, avançado).

Além disso, ele envia a nota dos alunos para os Agentes Tutor (um agente para cada aluno)

Segundo a nota que foi obtida por cada um dos alunos, o agente insere as informações no banco de dados do AVA, através do artefato, para que sejam mostrados (disponibilizados) aos alunos, os recursos e atividades levando em conta o perfil deles. Isto é feito na execução do plano *disponibiliza_recursos_atividades*. O objetivo do Agente Tutor é, tendo a informação da nota perfil do aluno, verificar a mudança no desempenho dele e enviar mensagens de incentivo ou de parabéns. Este agente tem um comportamento predominantemente reativo, pois atua após receber as mensagens do Agente Bedel. Porém, na estrutura geral do modelo, sua atuação pode, no futuro, ser estendida, levando em conta o cenário global do AVA e considerando a interação com agentes Bedel de diversas disciplinas, bem como outros agentes Tutor.

O Agente Bedel é considerado também um agente predominantemente reativo, pois inicia a execução dos planos em resposta a um evento específico do sistema, mas também apresenta um comportamento

proativo, por exemplo, na execução dos planos onde ele utiliza o conhecimento que guarda nas suas crenças para desencadear outros planos.

5.1.2. Modelo do aluno

Para (SILVEIRA, 1996)

Como todo modelo, um modelo de aluno é concebido para prover informações sobre o objeto modelado, que é o aluno. Neste caso, o estudante, em particular, que esteja usando um sistema computacional de aprendizagem. O sistema utiliza o modelo de aluno para ajudar a determinar as ações apropriadas ao estudante.

O conteúdo básico do modelo do aluno consiste na representação do seu conhecimento sobre o domínio e na especificação de seus objetivos. Pode ser acrescentado, ainda, as intenções, os planos, as atitudes do aluno e os procedimentos de inferência relevantes para a aplicação. (SILVEIRA, 1998)

O modelo do aluno contém informações do estado cognitivo do aluno, seu estilo cognitivo, habilidades e dificuldades. (ANDRADE, 2003)

O banco de dados do AVA tem as informações referentes ao aluno, tais como dados pessoais, dados de desempenho e interação do aluno no sistema. O modelo do aluno, neste trabalho, é utilizado para obter as informações necessárias dos alunos, para a execução dos diferentes planos por parte dos agentes.

Toda interação do aluno no ambiente é guardada no banco de dados em forma de *log*. Da mesma forma, o desempenho do aluno em cada uma das atividades e tarefas é armazenado no banco de dados e atualizado constantemente, a cada acesso do aluno, fornecendo um rico material para a atuação dos agentes.

O modelo do aluno é representado pelo id do aluno, as notas dele nas diferentes atividades propostas pelo professor e a informação de acesso a estas atividades (log). Este modelo leva em consideração, principalmente, o desempenho do aluno. Os agentes têm acesso a estas informações através do banco de dados do AVA, que guarda todos os dados da interação do aluno com o ambiente e de sua participação na disciplina.

Os agentes acessam o banco quantas vezes forem necessárias, para atualizar as informações do sistema. Eles compartilham a informação do banco. O Agente Tutor atualiza as crenças que ele tem do aluno e, se precisar, mostra alguma mensagem para ele sobre seu desempenho. O Agente Bedel obtém do banco de dados a configuração que o professor fez dos recursos e atividades da disciplina, configura a visualização deles e verifica se as tarefas foram avaliadas para mandar mensagem ao Agente Tutor que ao receber a mensagem atualiza os dados de perfil dos alunos.

5.2. COMO FUNCIONA O MODELO PROPOSTO?

No modelo proposto, formado pelos atores professor e aluno e pelos agentes Tutor e Bedel, os alunos são agrupados em três perfis diferentes, segundo o desempenho (notas) nas tarefas e o acesso deles nos diferentes recursos (material didático). Estes perfis são básico, intermediário e avançado.

Esta separação em grupos leva em conta a “nota perfil” de cada um dos alunos, que é calculada da seguinte forma:

Pega-se a nota da última atividade avaliada pelo professor e soma-se a ela a nota dos acessos às leituras que são pre-requisito da atividade. Se o aluno acessa a leitura ganha 2 pontos na nota da atividade, caso ele não acesse ganha 4 pontos. Esta diferença é dada para aumentar a possibilidade do aluno que não acessa a leitura ter uma nota_perfil mais alta e acesse a próxima tarefa de nível superior ao do perfil que ele pertenceria se tivesse uma nota menor, incentivando-o a procurar a leitura antes de realizar as atividades futuras.

Após isto, é feita a média dos valores do campo `nota_perfil` de todos os alunos.

O menor valor considerado para a média do perfil é 6 e o valor máximo considerado para a média do perfil é 8.

A nota perfil é calculada da seguinte forma:

$$\text{Nota_perfil_aluno} = \frac{\text{Soma_notas} + (\text{Nota_atividade} + \text{Nota_acessos})}{\text{Nro_calculo}}$$

A soma das notas é calculada multiplicando o valor da última nota perfil do aluno com o número de vezes que a nota perfil foi calculada (`último_nro_cálculo`), antes do cálculo corrente.

A `nota_atividade` é a nota da última atividade avaliada pelo professor, que “ativou” o cálculo da nova `nota_perfil`.

A `nota_acessos` é uma pontuação que é somada à `nota_atividade` dependendo se o aluno leu ou não leu o conteúdo que é pré-requisito dela.

O `nro_calculo` é a quantidade de vezes que está sendo calculada a nota perfil, incluindo o cálculo corrente. Este valor é igual ao `último_nro_cálculo + 1`.

A `última_nota_perfil` é a nota perfil do aluno na posição [`último_nro_cálculo`].

O aluno pertence ao perfil médio se a sua `nota_perfil` está entre 0.5 mais ou menos do que a média da `nota_perfil` da turma, por exemplo, se a média da `nota_perfil` for 7.5 ele estará no perfil médio se tiver sua `nota_perfil` entre 7 e 8. O aluno que tiver uma nota maior com uma diferença superior a 0.5 da nota média terá o perfil avançado e o aluno que tiver uma nota menor com diferença superior a 0.5 terá o perfil básico.

Estes valores foram utilizados devido ao fato das notas atualmente ficarem no intervalo de 0 a 10 e os alunos serem considerados aprovados com notas maiores ou muito perto de 6.

O Agente Bedel é o agente da disciplina que recebe as informações inseridas pelo professor na hora de configurar o sistema de tutoria, por meio do bloco criado no ambiente. Com estas informações, o Bedel sabe o comportamento que deve seguir para configurar a disciplina de forma personalizada segundo o perfil de cada aluno, inserindo as informações necessárias no banco de dados. Além disso, se encarrega de verificar as atualizações do professor nas notas das avaliações, calcula a nota perfil dos alunos, calcula a média das notas perfil, atualiza o banco de dados com estas notas e se comunica com o Agente Tutor para informá-lo do desempenho dos alunos.

O Agente Tutor é o agente que entra em contato com o aluno, ele obtém informações enviadas pelo Agente Bedel, sobre o desempenho dos alunos, cada vez que é atualizada a planilha de notas e se comunica com eles para incentivá-los ou parabenizá-los segundo o que for preciso no momento.

O modelo proposto utiliza o modelo de aluno, e tutores inteligentes (agentes Bedel e Tutor) da seguinte forma:

Etapa 1: Preparação do ambiente

- O professor insere os recursos e atividades normalmente no ambiente, utilizando as ferramentas disponíveis no AVA da forma como ele já está acostumado.
- O professor configura o agente utilizando a interface do ambiente indicando:

- Nível de dificuldade (básico, intermediário, avançado, geral) de cada um dos recursos e atividades que ele inseriu anteriormente no AVA. O nível de dificuldade geral é selecionado quando o professor quer que o recurso/atividade seja mostrado a todos os alunos por igual.
- Quais são os recursos e atividades por onde o aluno deve iniciar o processo. A primeira leitura (recurso) e a primeira atividade são mostradas para todos os alunos, portanto, o professor precisa indicar quais são elas.
- As dependências entre atividades e recursos.
- Com estas informações o Agente Bedel, que é o agente da disciplina, tem conhecimento dos recursos e atividades desta e sabe como deve ser desenvolvido o curso para cada tipo de estudante.
- O professor visualiza o grafo de dependências que é gerado pelo sistema, depois de concluída a etapa anterior.

Etapa 2:

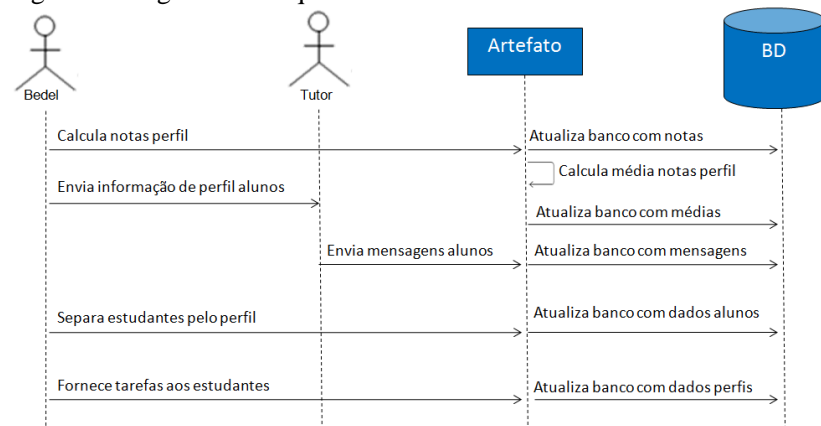
- O professor atribui as notas nas atividades propostas, depois de realizadas pelos alunos, fazendo com que a planilha de notas para todos os alunos na disciplina esteja atualizada.
- O Agente Bedel verifica que foram dadas as notas para todos os alunos em determinada atividade e calcula a nota perfil dessa atividade utilizando a nota dada pelo professor e os logs de acesso aos recursos sinalizados como pré-requisitos da atividade, na configuração inicial do professor. O Agente Bedel faz o cálculo da média da nota perfil de todos os alunos e com isso os alunos são separados pelo perfil nos grupos básico, intermediário ou avançado, segundo a nota perfil deles, sendo que quem tem a nota média fica no perfil intermediário, quem tem nota abaixo da média fica no perfil básico e quem tem nota acima da média fica no perfil avançado.

Etapa 3:

- O Agente Bedel verifica o perfil ao qual pertencem os alunos e mostra as seguintes atividades segundo o perfil deles.
- Os alunos acessam as atividades de forma personalizada, segundo a nota obtida em atividades anteriores e o acesso aos recursos (leituras).
- Cada vez que o Agente Bedel calcula a nota perfil ele atualiza no banco de dados o perfil corrente do aluno, que pode passar de básico para intermediário ou avançado e viceversa, durante o tempo que a disciplina é oferecida.
- O processo se repete desde a etapa 2.

Na figura 4 pode-se ver o diagrama de sequência que mostra as interações entre os agentes, Bedel e Tutor, o artefato e o banco de dados.

Figura 4. Diagrama de sequência



6. IMPLEMENTAÇÃO DO MODELO

O modelo integra conceitos de arquiteturas de sistemas de tutores inteligentes com ambientes virtuais de aprendizagem de uso consolidado como o Moodle, que não são adaptativos por si só, e que podem ser potencializados com técnicas de inteligência artificial, dando como resultado ambientes inteligentes de aprendizagem que se mostram adaptativos e mais adequados para a implementação de metodologias de ensino desafiadoras para o aluno.

O uso de agentes na implementação deste modelo é importante pela capacidade dos agentes de se adaptar às mudanças no ambiente, mostrando os recursos e atividades aos alunos de forma personalizada, segundo o desempenho deles na disciplina, levando em conta as configurações iniciais do professor. A habilidade dos agentes para adaptar-se às mudanças no ambiente é um fator importante nesta integração, pois os dados dos alunos mudam constantemente.

O conceito de adaptação neste trabalho está relacionado com a capacidade dos agentes de responder aos estímulos do ambiente de diferentes formas. Dependendo das informações que eles recebem deste, eles conseguem executar diversos planos.

Para a implementação do agente, é utilizada a ferramenta *Jason*, que é um interpretador para uma versão estendida do *AgentSpeak*, linguagem de programação orientada a agentes, implementado em Java. A característica do *AgentSpeak* é definir o *know-how* (conhecimento sobre como fazer as coisas) de um programa, na forma de planos (Bordini, 2007), como os utilizados pelos agentes BDI convertendo os desejos em intenções.

Um dos aspectos mais interessantes do *AgentSpeak* é que inspira-se e baseia-se em um modelo de comportamento humano que foi desenvolvido por filósofos. Esse modelo é chamado de modelo *belief-desire-intention* (BDI). A linguagem interpretada por *Jason* é uma extensão da *AgentSpeak*, baseada na arquitetura BDI. Um dos componentes da arquitetura de agentes é uma base de crenças, e um

exemplo do que o interpretador faz constantemente, sem ser especificamente programado, é perceber o ambiente e atualizar a base de crenças em conformidade com isso (Bordini, 2007).

Os ambientes virtuais de ensino-aprendizagem (AVEAs) são projetados para viabilizar o processo de construção de conhecimento. Diferentemente do software convencional, que busca facilitar a realização de tarefas pelo usuário, os ambientes para aprendizagem incorporam a complexidade de flexibilizar diferentes formas do usuário (aluno), fazer relações, apreender e praticar conteúdos, colaborar. Estes ambientes são utilizados por alunos de perfis cognitivos diversos (Boff, 2008).

Segundo a pesquisa feita pelo AL-AJLAN (2008), o Moodle tem uma arquitetura, implementação, interoperabilidade, e internacionalização boas, além de ter uma comunidade muito forte, é gratuito e sua acessibilidade é média. Tem quase a pontuação máxima em funcionalidades esperadas para uma plataforma de *e-learning*, e tem a melhor classificação na categoria de adaptação. Além disso, recursos de personalização e adaptabilidade estão presentes no Moodle.

Por ser o ambiente virtual de aprendizagem com recursos de adaptabilidade reconhecidos, para esse trabalho, o ambiente virtual de aprendizagem utilizado é o Moodle.

6.1. MOODLE

Acrônimo de *Modular Object-Oriented Dynamica Learning Environment*, o Moodle é um Sistema de Gerenciamento de Cursos (*Course Management System – CMS*), conhecido também como *Learning Management System (LMS)* ou Ambiente Virtual de Aprendizagem. Ele possui características que permitem seu uso por uma grande quantidade de alunos, assim como por grupos pequenos. (MOODLE ORG, 2012)

O Moodle é um *software* livre usado por diferentes instituições, desde escolas primárias até universidades, como apoio no processo de ensino-

aprendizagem, tanto em cursos de educação a distância quanto em cursos presenciais.

A comunidade de desenvolvedores do Moodle reúne programadores, desenvolvedores, administradores de sistema, professores, designers e usuários do mundo inteiro. Segundo o seu site principal moodle.org (2012), o Moodle está disponível em mais de 75 línguas diferentes e é usado em mais de 220 países, com mais de 70.300 sites registrados.

Esses dados confirmam a preferência pelo uso do Moodle, software de distribuição gratuita, constantemente atualizado pelos colaboradores da comunidade, aumentando a quantidade de módulos disponíveis para a implementação de diversas funcionalidades, assim como melhorando módulos existentes.

A versão do Ambiente Virtual de Aprendizagem Moodle utilizada para este trabalho é a 2.2, onde está disponível o recurso de condição de tarefas, que permite disponibilizar conteúdo e atividades com uma restrição. Este recurso deve ser ativado pelo administrador do Moodle nas configurações avançadas do ambiente, habilitando as opções “Habilitar rastreamento de conclusão” e “Habilitar acesso condicional”. Além disto, nas configurações do curso, no tópico “Progresso do aluno” a opção de acompanhamento de conclusão deve ser habilitada pelo professor.

Estando habilitado este recurso, as tarefas podem ser disponibilizadas somente para os alunos que cumprirem os pré-requisitos definidos. Estes podem ser: uma nota específica em uma atividade, ou a visualização ou nota em um recurso.

Neste trabalho, a disponibilização dos recursos e atividades é feita levando em conta o desempenho do aluno e o seu acesso no sistema, para isto, ao invés de utilizar as opções de nota de atividade e visualização ou nota de recurso, é feita a disponibilização por aluno, dependendo da nota_perfil dele, calculada segundo seu rendimento e participação na disciplina. As informações entre o sistema de agentes e o ambiente virtual de aprendizagem são trocadas por meio do banco de

dados do ambiente, que contém as informações sobre os pré-requisitos das tarefas e recursos, definidos pelo professor na hora de configurar o Agente Bedel.

6.1.1. Banco de dados do Moodle

O Moodle conta com um banco de dados que guarda todos os dados do ambiente, desde informação dos usuários e logs de atividades até configurações internas da instalação dele. O número de tabelas criadas na instalação padrão do ambiente é de 280 tabelas. Neste trabalho são utilizadas as seguintes tabelas:

- Role
- Role_assignments
- Context
- Log
- Course_modules
- Course
- Assignment
- Grade_items
- Grades_grades
- Modules

Além disso, foram criadas tabelas específicas necessárias para o desenvolvimento do modelo do agente, para guardar os dados do perfil do aluno (básico, intermediário, avançado) e a nota de perfil dele. Essas tabelas são:

- Tutor_dependencia
- Tutor_tarefas_avaliables
- Tutor_nota_perfil
- Tutor_media_perfis
- Tutor_perfis
- Tutor_profile_availability
- Tutor_rec_at_perfil

6.2. AMBIENTE DE DESENVOLVIMENTO JASON

Jason é uma implementação particular da linguagem de programação de agentes *AgentSpeak*, planejada para desenvolver sistemas multiagente e inspirada e baseada no modelo B.D.I., descrito anteriormente.

Um dos componentes da arquitetura dos agentes é a base de crenças, que será constantemente atualizada pelo interpretador, que percebe o ambiente, sem precisar estar programado para isso. Outro componente

importante são os objetivos dos agentes, que serão alcançados na execução dos planos.

Jason usa uma linguagem de programação de agentes baseada em lógica dentro de uma plataforma Java e requer o uso de Java para certas tarefas avançadas. (BORDINI, 2007)

No *Jason*, um agente é caracterizado por suas crenças, planos e eventos que são recebidos do ambiente ou gerados internamente. Um plano em *Jason* é designado para um evento específico em um contexto de crença. (DASTANI, 2008).

Os agentes foram desenvolvidos utilizando a ideia de artefatos e foram implementados com o CArTAgO (*Common ARTifact infrastructure for AGents Open environments*), que é um *framework* para programar e executar ambientes virtuais para sistemas multiagentes.

A implementação dos agentes com o ambiente (*environment*) sendo um artefato facilitou muito o acesso ao banco de dados do ambiente virtual de aprendizagem. O fato do artefato ser implementado em java, utilizando a linguagem sql no código fez com que a implementação dos agentes fique mais simples.

Os agentes têm os planos deles e dentro destes planos se comunicam com o artefato, que executa os métodos para os diversos propósitos dos agentes e quando precisa, devolve alguma informação relevante para eles.

A implementação dos agentes fica mais clara e todo o acesso ao banco de dados é centralizado no artefato.

6.3. ETAPAS DE DESENVOLVIMENTO:

O desenvolvimento deste modelo foi realizado em 4 etapas:

- Desenvolvimento do bloco tutor para o ambiente Moodle
- Desenvolvimento do cálculo de perfil do aluno (Agente Bedel)

- Desenvolvimento do código de disponibilização dos recursos e atividades (Moodle).
- Desenvolvimento do Agente Tutor – mensagens de *feedback* para o aluno.

6.3.1. Desenvolvimento do bloco tutor

Foi desenvolvido um bloco no moodle para que o professor configure o agente.

Este bloco foi desenvolvido seguindo o padrão de programação do Moodle para a criação de blocos, em linguagem PHP e acessando o Banco de Dados do Moodle utilizando a linguagem SQL.

O nome do bloco foi definido como “Tutor”. Neste o professor define o nível dos diferentes recursos e atividades (básico, intermediário ou avançado), escolhe o recurso e atividade iniciais (Figura 5), os quais ficam disponíveis da mesma forma para todos os alunos e são a base para o cálculo inicial da nota perfil do aluno e, finalmente, escolhe os diferentes pré-requisitos (Figura 6) para cada atividade e/ou recurso da disciplina.

Cabe lembrar que o professor, antes de fazer uso deste bloco no Moodle deve utilizar os recursos e ferramentas disponíveis no Moodle da forma usual para publicar os conteúdos e as atividades referentes a cada tópico da matéria. A diferença é que para que o sistema possa se tornar adaptativo, o professor deve propor o maior repertório de recursos e atividades possível em um mesmo tópico para que ele possa, posteriormente, definir os diferentes níveis e o sequenciamento entre cada um dos itens postados.

Ao finalizar a configuração do agente, por meio do bloco, o professor visualiza o grafo de dependências (Figura 7) gerado após a definição de todos os pré-requisitos.

O recurso e a atividade iniciais podem ser editados, caso o professor precise modificá-los, porém, todos os pré-requisitos selecionados após a

escolha destes, serão descartados, pois tendo um novo recurso e uma nova atividade iniciais a ordem dos pré-requisitos muda. Por outro lado, nesta versão do modelo, os pré-requisitos selecionados para cada atividade e/ou recurso, podem ser editados, sem ter dados descartados.

Figura 5. Seleção do recurso e atividade iniciais

Selecione um recurso e uma atividade para iniciar o estudo:

Recursos:

- ☒ Leitura_1
- ☐ Leitura_2_Nível_Básico
- ☐ Leitura_2_Nível_Intermediário
- ☐ Leitura_2_Nível_Avançado
- ☐ Leitura_3_Geral
- ☐ Leitura_4_Nível_Básico
- ☐ Leitura_4_Nível_Intermediário
- ☐ Leitura_4_Nível_Avançado

Atividades:

- ☒ Atividade_1
- ☐ Atividade_2_Nível_Básico
- ☐ Atividade_3_Geral
- ☐ Atividade_2_Nível_Intermediário
- ☐ Atividade_2_Nível_Avançado
- ☐ Atividade_4_Nível_Básico
- ☐ Atividade_4_Nível_Intermediário
- ☐ Atividade_4_Nível_Avançado

Selecione

6.3.2. Desenvolvimento do cálculo de perfil do aluno (Bedel)

A programação do agente foi realizada utilizando a IDE Eclipse, com o plugin do Jason. Os agentes foram desenvolvidos com artefatos implementados em CartAgO, na linguagem Java.

Figura 6. Seleção de pré-requisitos

Recurso e atividade iniciais

Recurso Inicial: Leitura_1

Atividade Inicial: Atividade_1

Editar

Atenção: Editar os recursos e atividades iniciais implica em apagar no banco de dados todos os pré-requisitos definidos até o momento.

Selecione um recurso ou uma atividade para informar seus pré-requisitos

Recursos e atividades:

Leitura_2_Nível_ ▾

Selecione os pré-requisitos

Recursos: ☐ Leitura_1
☐ Leitura_2_Nível_Intermediário
☐ Leitura_2_Nível_Avançado
☐ Leitura_3_Geral
☐ Leitura_4_Nível_Básico
☐ Leitura_4_Nível_Intermediário
☐ Leitura_4_Nível_Avançado

Atividades: ☐ Atividade_1
☐ Atividade_2_Nível_Básico
☐ Atividade_3_Geral
☐ Atividade_2_Nível_Intermediário
☐ Atividade_2_Nível_Avançado
☐ Atividade_4_Nível_Básico
☐ Atividade_4_Nível_Intermediário
☐ Atividade_4_Nível_Avançado

Fóruns:

Obs: O fórum de notícias não pode ser usado como pré-requisito

Voltar

Seleciona

A implementação dos agentes está dividida basicamente nos seguintes arquivos:

- jasonTutor.mas2j: Arquivo de definição do sistema multiagente. Aqui se especifica a infraestrutura centralizada, ambiente CartAgO e os agentes Tutor e Bedel.
- BD_Artifact.java: Arquivo com o código do artefato, que faz a conexão entre os agentes e o banco de dados do ambiente virtual de aprendizagem.

- tutor.asl: Arquivo com a implementação do Agente Tutor, que é o agente que entra em contato com o aluno.
- bedel.asl: Arquivo com a implementação do Agente Bedel, agente da disciplina, que segue as configurações definidas pelo professor no bloco criado no AVA, para mostrar aos alunos os recursos e atividades dependendo o desempenho deles.

Além destes foram desenvolvidos alguns arquivos secundários de associação de dados, criados para facilitar o acesso à informação recebida nas consultas ao banco de dados.

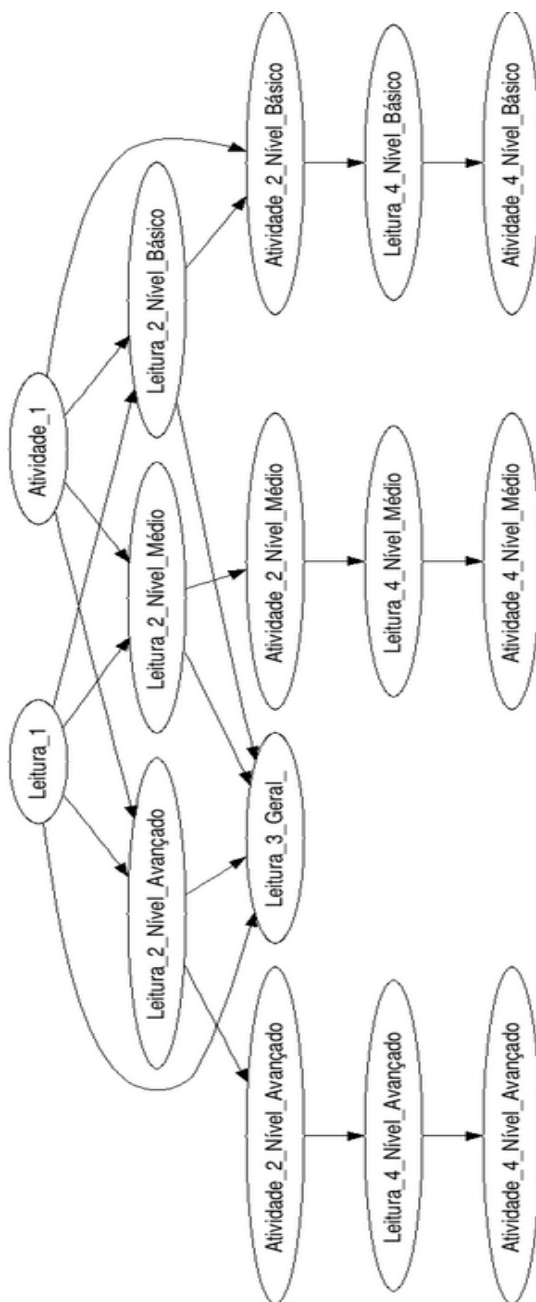
Nesta etapa do desenvolvimento foi implementado o arquivo de definição do sistema multiagente jasonTutor.mas2j, o arquivo com o código do artefato BD_Artifact.java, e o arquivo do agente bedel.asl, ambos com os métodos necessários para que o agente possa calcular a nota perfil do aluno.

O arquivo BD_Artifact.java contém, nesta etapa do desenvolvimento, os métodos de acesso e atualização do banco de dados, além dos métodos chamados pelo Agente Bedel.

O arquivo bedel.asl contém, nesta etapa do desenvolvimento, os planos para ativar o artefato, verificar o primeiro acesso do professor, verificar a configuração dos recursos e atividades para ficar ocultos aos alunos (com exceção dos iniciais), verificar se as atividades já passaram da data de entrega, verificar se estas atividades já foram avaliadas, calcular a nota de perfil dos alunos, calcular a média das notas de perfil dos alunos e calcular os valores dos perfis básico, intermediário e avançado para separar os alunos segundo estes perfis. Além disso, ele inicia com as crenças para verificar se o professor acessou ou não o ambiente pela primeira vez, para verificar se os recursos e atividades, após a configuração do tutor, ficaram ocultos para os alunos e para verificar que o artefato banco foi ativo.

Estas crenças são atualizadas nos planos para verificar o login do professor, verificar o olho fechado e para ativar o artefato, respectivamente.

Figura 7. Grafo de dependências



O Agente Bedel começa a execução dos planos com um gatilho gerado cada 24 horas, o agente, cada dia, começa com a verificação da data final das tarefas, se alguma delas passou da data de entrega ele verifica se a tarefa foi avaliada e, caso tenha alguma avaliação, ele executa o plano para calcular a nota perfil dos alunos, após isto ele calcula a média perfil e os valores de cada um dos perfis (básico, intermediário e avançado).

Durante a execução destes planos ele vai atualizando as crenças para saber se existe uma atividade para avaliar e para saber se a atividade já está avaliada (`existeAtividadeParaAvaliar`, `existeAtividadeAvaliada`).

Neste trabalho, a implementação do Agente Bedel no protótipo foi feita pensando em uma disciplina só, no entanto, o modelo foi pensado para ter uma instância do Agente Bedel para cada disciplina, levando em conta um ambiente virtual funcionando em toda sua capacidade, com um número grande de disciplinas, professores e alunos.

6.3.3. Desenvolvimento do código de disponibilização dos recursos e atividades (Moodle)

Para a disponibilização dos recursos e atividades segundo o perfil do aluno, foi criada a tabela `tutor_profile_availability` no banco de dados, para guardar as informações das notas mínima e máxima do perfil intermediário para que, verificando este dado, possam ser disponibilizados os recursos e atividades, segundo o perfil dos alunos.

Além desta tabela, o arquivo `conditionlib.php` precisa ser modificado no método `“is_available”`, inserindo o código necessário para que os recursos e as atividades sejam mostradas no ambiente de forma personalizada para cada aluno, pegando as informações inseridas pelo Agente Bedel no banco de dados, após ter feito o cálculo da nota perfil, da média perfil e as notas dos perfis básico e avançado.

Nesta etapa, também, é atualizado o arquivo `bedel.asl`, inserindo os métodos que permitem a comunicação entre ele e o Agente Tutor. Enviando as informações dos alunos.

6.3.4. Desenvolvimento do Agente Tutor – mensagens de feedback para o aluno

Nesta etapa é feito o desenvolvimento do código do Agente Tutor, que é o agente que se comunica com o aluno, a implementação deste agente é feita no arquivo tutor.asl. O Agente Tutor se encarrega de enviar ao aluno mensagens de incentivo ou de *feedback* positivo, dependendo o desempenho dele.

O Agente Tutor tem a crença da nota perfil do aluno, que é atualizada cada vez que o Agente Bedel manda a mensagem com a nova nota, prévia atualização da crença *atividadeAvaliada*. Após isto, ele executa o plano para atualizar a nota do aluno. E, segundo a situação do aluno, o Agente Tutor envia uma mensagem para ele, parabenizando-o ou incentivando-o a melhorar. Para que o Agente Tutor possa enviar mensagens para o aluno, este precisa ter uma conta de usuário do ambiente virtual de aprendizagem.

Neste trabalho a implementação do Agente Tutor, para efeitos do protótipo, foi feita pensando em um aluno só. Porém, na ideia geral do modelo, existe um Agente Tutor para cada aluno matriculado na disciplina.

De acordo com o modelo proposto, o Agente Tutor receberia a informação dos Agentes Bedel de cada uma das disciplinas, e enviaria as mensagens ao aluno dependendo o seu desempenho, levando em conta as notas perfis em todas as disciplinas.

O envio da mensagem para o aluno é feita através do AVA, inserindo a mensagem na tabela adequada em seu banco de dados. O Agente Tutor, utilizando o artefato, atualiza as tabelas *message* e *message_read* do AVA. Dessa forma o aluno visualiza a mensagem, na próxima vez que acessa o ambiente, devido à atuação da ferramenta de envio de mensagens ao aluno, do próprio AVA.

As mensagens enviadas pelo Agente Tutor podem ser de incentivo quando o aluno teve um desempenho baixo em relação à atividade

anterior ou de parabéns quando o aluno melhorou, conforme mostrado na Tabela 1. Este conjunto de mensagens pode, facilmente, ser estendido para o caso de uma atuação mais abrangente do Agente Tutor.

Tabela 1. Mensagens de incentivo do Agente Tutor

Incentivo	Avançado para Intermediário	<p>Olá, aluno!</p> <p>O seu desempenho na última atividade podia ter sido melhor!</p> <p>Vamos nos esforçar um pouco mais!?</p> <p>Eu sei que você é capaz!</p>
	Avançado para Básico	<p>Olá, aluno!</p> <p>O seu desempenho teve uma pequena queda, mas não desanime! É necessário que se esforce um pouco mais a próxima vez!</p> <p>Você tem capacidade! Você vai conseguir!</p>
	Intermediário para Básico	<p>Olá, aluno!</p> <p>Você precisa estudar um pouco mais para manter seu desempenho!</p> <p>O esforço sempre é recompensado!</p> <p>Vamos em frente!</p>
	Básico para Básico	<p>Olá, aluno!</p> <p>É necessário que você se esforce um pouco mais! Vamos tentar?</p> <p>Sei que você consegue!</p>

Tabela 2. Mensagens de parabenização do Agente Tutor

Parabenização	Intermediário para Avançado	Olá, aluno! Parabéns pelo seu desempenho! Você foi muito bem na última atividade avaliada! Continue assim!
	Básico para Avançado	Olá, aluno! Parabéns! Você conseguiu se superar! Continue assim!
	Intermediário para Intermediário	Olá, aluno! Parabéns! Seu desempenho continua bom, mas ainda podemos melhorar um pouco mais! Tenho certeza que você consegue muito mais!
	Avançado para Avançado	Olá, aluno! Parabéns! Você conseguiu se manter com um ótimo desempenho! Continue assim!

7. PROTÓTIPO DO MODELO

Para testar o modelo foi feita a adaptação no código do ambiente virtual de aprendizagem Moodle, para integrá-lo com os agentes que implementam os tutores inteligentes.

Para essa integração foram criadas novas tabelas no banco de dados do Moodle, todas iniciando com a palavra “tutor”:

- Tabela tutor_dependencia
- Tabela tutor_media_perfis
- Tabela tutor_nota_perfil
- Tabela tutor_perfil
- Tabela tutor_profile_availability
- Tabela tutor_rec_at_perfil
- Tabela tutor_tarefas_avaliables

A tabela tutor_dependencia guarda as informações de dependencia entre os diferentes recursos e atividades, informados pelo professor na hora de configurar o sistema de tutoria. Esta tabela está formada pelos seguintes campos:

- Curso_id → Identificador do curso
- Rec_ativ_id → Identificador do recurso ou da atividade cujo pré-requisito é selecionado.
- Pre_req_id → Identificador do recurso ou da atividade que é pré-requisito da atividade informada no campo anterior.

A tabela tutor_media perfil contém os dados da média de cada um dos perfis, depois de cada um dos cálculos feitos pelo agente. Esta tabela tem os seguintes campos:

- Curso_id → Identificador do curso
- Perfil_id → Identificador do perfil (básico, intermediário, avançado)

- Nota → Nota calculada para cada um dos perfis, tendo como base a média que é a nota do perfil intermediário. A nota do perfil básico é a média subtraindo 0.6 e a nota do perfil avançado é a média aumentada em 0.6.

Nro_calculo → É o número de vezes que foi feito o cálculo da média e as vezes que foram calculados o perfil básico e avançado. Esse número é o mesmo para os três perfis e aumenta cada vez que é calculada uma nova média. A tabela tutor_nota_perfil contém os dados da nota perfil de cada um dos alunos, calculada cada vez que uma atividade é avaliada. Esta tabela contém os seguintes campos:

- Curso_id → Identificador do curso
- Id_grade_item → Identificador do item de nota
- Aluno_id → Identificador do aluno
- Nota → Nota de perfil do aluno
- Nro_calculo → Número de cálculo realizado

A tabela tutor_perfil contém os identificadores de cada um dos perfis, básico, intermediário, avançado e geral, sendo que o geral é só utilizado quando o professor precisa que a atividade ou recurso fique disponível para todos os alunos, sem restrição nenhuma. E não é levada em conta na inserção de dados das outras tabelas que utilizam os perfis. Esta tabela só tem dois campos, o id e o nome do perfil.

A tabela tutor_profile_availability foi criada tendo como base a tabela course_modules_availability, que é a tabela que contém os dados de restrição para a disponibilização dos diferentes módulos do ambiente (recursos e atividades). A tabela tutor_profile_availability guarda a informação necessária para saber a nota mínima e máxima do perfil intermediário, para cada um dos recursos. Os campos desta tabela são:

- Course → Identificador do curso
- Coursemoduleid → Identificador do módulo (recurso ou atividade)
- Sourcecmid → Identificador do modulo do curso.
- Requiredcompletion → Informação do requisito de conclusão

- `Profilenro` → Identificador do perfil
- `Grademin` → Nota mínima do perfil intermediário para o módulo (recurso ou atividade)
- `Grademax` → Nota máxima do perfil intermediário para o módulo (recurso ou atividade)

A tabela `tutor_rec_at_perfil` contém a informação do perfil ao qual cada uma das atividades pertence, e que pode ser básico, intermediário, avançado ou geral. Esta tabela serve para que o tutor saiba que tipo de atividade mostrar para o aluno, dependendo o perfil dele. Os campos dela são:

- `Curso_id` → Identificador do curso
- `Rec_ativ_id` → Identificador do recurso ou da atividade
- `Perfil_id` → Identificador do perfil

A tabela `tutor_tarefas_avaliables` contém a lista de todos os itens que devem ser avaliados e a informação se estes foram avaliados ou não. Os campos desta tabela são:

- `Curso` → Identificador do curso
- `Id_grade_items` → Identificador dos itens de nota
- `Avaliada` → Boolean com a informação se o item foi ou não avaliado.

Além destas modificações no banco de dados foi criada a pasta `tutor`, dentro da pasta “*blocks*”, que contém as pastas dos diferentes blocos do Moodle, e dentro desta pasta foi implementado o novo bloco, com o nome “Tutor”. Os arquivos criados na implementação do bloco Tutor foram:

- `Block_tutor`
- `Edit_form`
- `Settings`
- `Tutor_Arvore`
- `Tutor_depend`
- `Tutor_form`

- Tutor_lista
- Tutor_prereq
- Version

Os arquivos `block_tutor`, `edit_form`, `settings` e `version` são arquivos padrão da criação de blocos no Moodle. Os outros arquivos foram criados durante a implementação, segundo a necessidade. Além destes arquivos, tem a pasta `lang`, também padrão na criação dos blocos, que contém a pasta “`en`” e dentro dela o arquivo `block_tutor.php` com as strings necessárias para o bloco.

Outra modificação necessária no Moodle é a implementação do código que controla a disponibilização dos diferentes recursos e atividades da disciplina. Este código é adicionado no arquivo `conditionlib.php` que se encontra na pasta `lib` da raiz do Moodle.

Todas estas mudanças foram realizadas para adequar o ambiente virtual de aprendizagem e integrá-lo com os agentes. A comunicação entre o ambiente e os agentes é feita pelo banco de dados, que é atualizado tanto pelo ambiente como pelo artefato BD, do lado dos agentes.

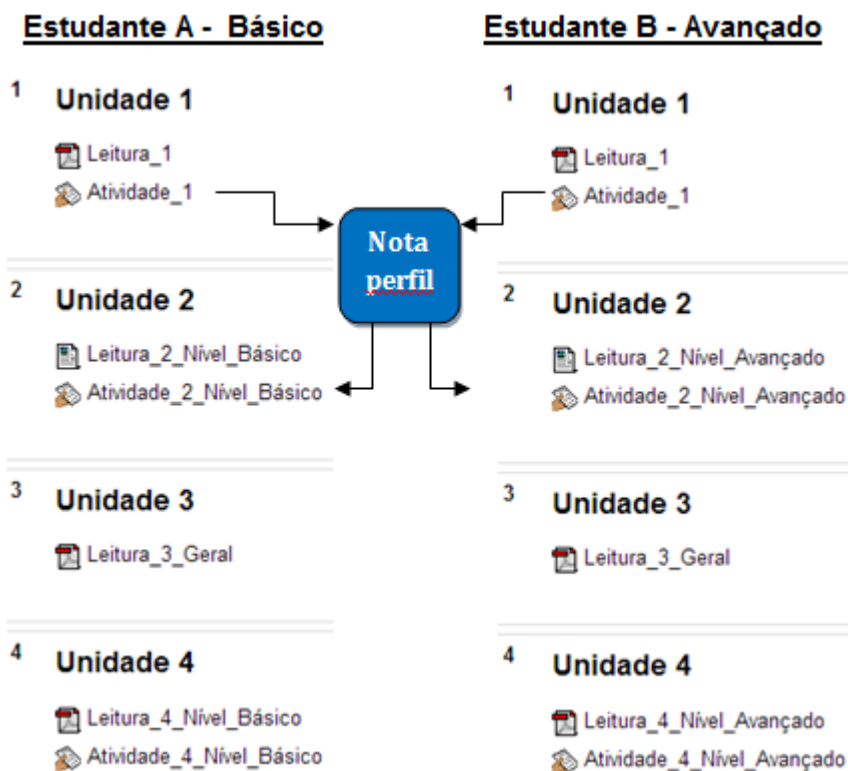
Os recursos e atividades no ambiente são visualizados de diferentes formas dependendo o perfil dos alunos, que podem passar do perfil básico para o intermediário ou avançado e viceversa, durante o tempo que a disciplina é ministrada.

Na figura 8, a seguir, é mostrada uma disciplina na visão de dois estudantes com perfil diferente. O estudante A tem o perfil básico e o estudante B tem o perfil avançado.

O primeiro recurso e a primeira atividade são mostrados para os dois alunos. Após a entrega da primeira atividade, é definido o perfil deles (pela nota perfil), e são mostrados os recursos e atividades segundo este.

Estes perfis são calculados cada vez que os estudantes obtêm uma nota nova nas atividades e podem mudar entre os três níveis (básico, intermediário, avançado). No exemplo mostrado os estudantes permaneceram no mesmo perfil durante toda a disciplina.

Figura 8. Visão dos estudantes



Os alunos visualizam só o que está disponível para eles, já o professor, tem acesso a todos os recursos e atividades da disciplina.

Na visão do professor (Figura 9) os recursos e atividades que têm pré-requisitos para serem mostrados aos alunos se mostram na cor cinza, pois eles não são disponibilizados da mesma forma para todos os alunos. Só os recursos e atividades iniciais ou com perfil “geral” se mostram

como disponíveis, pois eles aparecem no ambiente para todos os alunos sem distinção.

A configuração de perfil geral nos diferentes recursos ou atividades é usada pelo professor para disponibilizar material ou tarefas que ele considera necessários para a aprendizagem do conteúdo.

Figura 9. Visão do professor



8. CONSIDERAÇÕES FINAIS E TRABALHOS FUTUROS

Nesse estudo é feita uma solução inteligente para ambientes virtuais de aprendizagem, para ajudar aos professores na disponibilização de atividades e recursos de uma forma personalizada dependendo do desempenho do aluno e seu comportamento na disciplina.

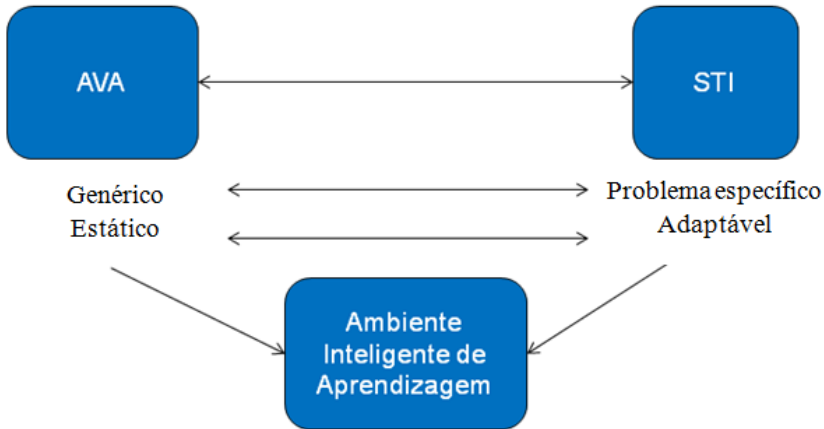
Os alunos são avaliados pela interação deles na disciplina e pelas notas obtidas nas tarefas, criando perfis diferentes para grupos de alunos com o mesmo comportamento. São disponibilizadas tarefas mais avançadas para alunos que tenham um melhor desempenho, possibilitando uma aprendizagem mais eficiente, explorando as habilidades dos alunos, mantendo um nível básico para o aprendizado do conteúdo da disciplina.

Trabalhos relacionados com ambientes virtuais de aprendizagem e adaptividade, no geral, diferenciam os alunos pelo estilo de aprendizagem, por exemplo, um aluno que aprende melhor com imagens do que com leituras com muito texto.

No primeiro trabalho correlato a adaptabilidade se baseia nas preferências dos alunos em relação ao material de estudo, onde o agente disponibiliza este material segundo as informações das preferências que ele tem do aluno. No segundo trabalho, a proposta é oferecer *layouts* diferentes aos alunos, levando em consideração preferências similares no que se refere a estilos de aprendizagem e escolhas de usabilidade. No último trabalho correlato se utilizam questionários de auto-avaliação criados adaptativamente no ambiente virtual de aprendizagem Moodle, e se utilizam ontologias para indexar os materias de aprendizagem.

Nesse trabalho os alunos se diferenciam pelo seu desempenho, levando em conta as notas obtidas, e pela sua participação (acessos) nos diversos recursos disponíveis na disciplina, criando um ambiente adaptativo, que atualiza de forma constante o perfil dos alunos, pois um aluno com perfil básico, ao final da disciplina pode ter um perfil médio o avançado. Essas mudanças de perfil podem ser estudadas e mostradas ao professor, em uma extensão deste modelo.

Figura 10. Ambiente inteligente de aprendizagem



A principal contribuição deste trabalho é adicionar as vantagens do ambiente virtual de aprendizagem ao sistema de tutores inteligentes e viceversa, criando um ambiente inteligente de aprendizagem (Figura 10).

Como trabalho futuro pode-se mencionar o aperfeiçoamento do bloco de configuração do tutor criado no moodle, no que se refere a níveis de dependência entre os recursos, para evitar dependências cíclicas.

Além disso, a implementação de relatórios, disponíveis para o professor, com o desempenho de todos os estudantes durante o semestre.

E a integração com repositórios de objetos de aprendizagem, com a finalidade de disponibilizar aos alunos objetos que estejam de acordo com seu nível de aprendizado, obtido com os dados do seu desempenho.

REFERÊNCIAS

AL-AJLAN Ajlan, ZEDANs Husein. *Why Moodle, in International Workshop on Future Trends of Distributed Computing System*. IEEE 2008.

ANDRADE, A. F.; VICARI, R. M. Construindo um ambiente de aprendizagem a distância inspirado na concepção sociointeracionista de Vygotsky. In: SILVA, M. (Org) *Educação Online*. São Paulo: Edições Loyola, 2003.

BAZIUKAITÉ Dalia. *Approach to an Adaptive and Intelligent Learning Environment, in Advances in Computer, Information, and Systems Sciences, and Engineering*, 399-466. Springer 2006.

BIANCONCINI Maria E. de A. Educação a distância na internet. *Educação e pesquisa*, São Paulo, v.29, n.2, p. 327-340, jul/dez. 2003. Disponível em: <http://www.scielo.br/pdf/ep/v29n2/a10v29n2.pdf>. Acesso em 28 Nov. 2011.

BORDINI Rafael, VIEIRA Renata, MOREIRA Alvaro F. Fundamentos de Sistemas Multiagentes. In: FERREIRA, C. E. (Ed.). *Jornada de Atualização em Informática (JAI'01)*. Fortaleza, Brasil: SBC, 2001. v. 2, cap. 1, p. 3-44

BRENER Walter, ZARNEKOW Rüdiger, WITTIG Hartmut. *Intelligent Software Agents – Foundations and Applications*. Springer-Verlag Berlin Heidelberg. 1998.

BRUSILOVSKY Peter. (1994). *Student model centered architecture for intelligent learning environment*. In *Proceedings of Fourth International Conference on User Modeling*. Disponível em: <http://www2.sis.pitt.edu/~peterb/papers/UM94.html>. Acesso em 04 Dez. 2011.

CARTAGO Disponível em: <<http://cartago.sourceforge.net/>>. Acesso em: 12 dez. 2012.

CATAPAN A. H; MALLMANN E. M., NUNES, I. K. C. e RONCARELLI, D. *Pedagogical Mediation and Virtual Environ of Teaching-Learning*. In: ICDE – 22ª Conferencia Mundial de Educação a Distância. Rio de Janeiro, 2006.

CONATI, C. *Intelligent tutoring systems: new challenges and directions. Paper presented at the Proceedings of the 21st international joint conference on Artificial intelligence*. 2009.

DASTANI Mehdi. *2APL: a practical agent programming language*. In *Autonomous Agent and Multi-Agent System*. 2008.

DILLENBOURG, P., HILARIO, M., MENDELSON, P., SCHNEIDER, D., BORCIC, B., *Intelligent Learning Environments*, Project No. 4023-2701, TECFA, University of Geneva, 1993.

DILLENBOURG, P. *Virtual Learning Environment*. 2000. Disponível em:

<<http://tecfa.unige.ch/tecfa/publicat/dil-papers-2/Dil.7.5.18.pdf>>.

Acesso em: 28 Nov. 2011.

FAGUNDES Mosser. Um ambiente para desenvolvimento de agentes B.D.I. Trabalho de conclusão de curso. Universidade Federal de Pelotas. 2004.

Disponível

em:

http://www.inf.ufsc.br/~silveira/INE602200/Artigos/TCC_Moser.pdf.

Acesso em 04 Dez. 2011.

FELTES Heloisa. *Semântica cognitiva e modelos culturais: perspectivas de pesquisa*. Disponível em:

http://www.pessoal.utfpr.edu.br/paulo/semantica%20cognitiva_introducao.pdf . Acesso em 14 Dez. 2011.

FRASSON Claude, MENGELLE Thierry, AIMEUR Esma. *Using pedagogical agents in a multi-strategic intelligent tutoring system*. In *Proceedings of the A I-ED '97 Workshop on Pedagogical Agents*, pages 40-47. 1997.

FRIGO L. B., POZZEBON E., BITTENCOURT G. O Papel dos Agentes Inteligentes nos Sistemas Tutores Inteligentes, World Congress on Engineering and Technology Education, São Paulo, Brasil. 2004.

GIL, A.C. Como elaborar projetos de pesquisa. São Paulo: Atlas. Links, p. 207, 1991.

GIRAFFA Lúcia M. M. Uma arquitetura de tutor utilizando estados mentais. Tese (Doutorado em Ciências da Computação) – Instituto de Informática, UFRGS, Porto Alegre. 1999. Disponível em: <http://www.lume.ufrgs.br/bitstream/handle/10183/17620/000269142.pdf?sequence=1>. Acesso em 04 Dez. 2011.

GIRAFFA Lúcia M. M, VICCARI R. M., *The use of agents techniques on intelligent tutoring systems. In Computer Science, 1998. SCCC '98. XVIII International Conference of the Chilean Society*

GRAF Sabine, KINSHUK. *Analysing the Behaviour of Students in Learning Management Systems with Respect to Learning Styles*, in Springer-Verlag Berlin Heidelberg, 2008.

HUBNER Jomi F. Um Modelo de Reorganização de Sistemas Multiagentes. PhD thesis, Universidade de São Paulo, Escola Politécnica, 2003. Disponível em: <http://www.das.ufsc.br/~jomi/pubs/Hubner-tese.pdf>. Acesso em 04 Dez. 2011.

JAQUES, Patrícia Augustin ; VICARI, Rosa Maria . Estado da Arte em Ambientes Inteligentes de Aprendizagem que Consideram a Afetividade do Aluno. Informática na educação, UFRGS: Porto Alegre, v. 8, n. 1, p. 15-38, 2005.

JOHNSON W. Lewis, SHAW Erin, GANESHAN Rajaram. *Pedagogical agents on the web. In Proceedings of the third annual conference on autonomous agenst. New York, NY, USA, 1999.*

LIMA, R.D. e Rosatelli, M.C. (2004) “Um Sistema Tutor Inteligente para um Ambiente Virtual de Ensino Aprendizagem”. Anais do WIE. 2003.

MOODLE Org. Disponível em: <<https://moodle.org/>>. Acesso em: 10 dez. 2011.

MORAN, José Manuel. Novos caminhos do ensino à distância. *Informe CEAD - Centro de Educação à Distância*. SENAI. Rio de Janeiro, Ano 1, n. 5, out/nov/dez 1994, p. 1-3.

OLIVEIRA Cláudio L. V. AutoExplC: sistema tutor inteligente para auxílio ao ensino da linguagem “C” baseado na aprendizagem por auto-explicação de exemplos - Campinas: PUC-Campinas, 2005.

OMICINI, A., RICCI, A., & VIROLI, M. (2008). *Artifacts in the A&A meta-model for multi-agent systems. Autonomous Agents and Multi-Agent Systems*, 17(3), 432–456.

PADGHAM Lin, WINIKOFF Michael. *Developing intelligent agent systems – a practical guide*. Wiley. 2004.

PEREIRA Alice T C, SCHMITT Valdenise, ÁLVARES Maria R C Dias. Ambientes Virtuais de Aprendizagem. Livraria Cultura, 2007. Disponível em: <http://www.livrariacultura.com.br/imagem/capitulo/2259532.pdf>. Acesso em 27 Nov. 2011.

RAO Anand S., GEORGEFF Michael P. *BDI Agents: From Theory to Practice*. In *Proceedings of the First International Conference of Multiagents Systems*. AAAI, 1995. Disponível em: <https://www.aaai.org/Papers/ICMAS/1995/ICMAS95-042.pdf>. Acesso em: 04 Dez. 2011.

ROMANÓ Rosana Schwansee. A utilização de ambientes virtuais para a aprendizagem colaborativa no ensino fundamental. Dissertação (mestrado) - Universidade Federal de Santa Catarina, Centro Tecnológico. Programa de Pós-Graduação em Engenharia de Produção. Bibliografia : f. 98-100. 2002.

RONCARELLI, Dóris. Nas Asas de Ícaro: O Reomodo do Fazer Pedagógico. Construindo uma taxionomia para escolha de Ambiente Virtual de Ensino-Aprendizagem – AVEA. Dissertação de Mestrado. Florianópolis: UFSC/PPGE, 2007a.

RONCARELLI, D., Mallmann, E. M. e CATAPAN, A. H. EaDList: uma ferramenta para escolha de um Ambiente Virtual de Ensino-Aprendizagem. In: 13º Congresso Internacional de Educação a Distância, Curitiba. 2007b.

RUSSEL Stuart, NORVIG Peter. *Artificial Intelligence: A Modern Approach*. Prentice-Hall, Inc. 1995.

SANTOS. Edméa Oliveira. Ambientes virtuais de aprendizagem: por autorias livre, plurais e gratuitas. In: Revista FAEBA, v.12, no. 18.2003.

SANTOS, Cássia Trojahn; FROZZA, Rejane; DAHMER, Alessandra; GASPARY, Luciano Paschoal. Dóris – Um agente de acompanhamento pedagógico em sistemas tutores inteligentes. In: SBIE'01 SIMPÓSIO BRASILEIRO DE INFORMÁTICA NA EDUCAÇÃO, 12.,2001, Vitória-ES. Anais. Vitória: UFES, 2001.

SILVA Edna L, MENEZES Estera M. Metodologia da pesquisa e elaboração de dissertação. 4. ed. rev. atual. – Florianópolis: UFSC, 2005. 138p.

SILVEIRA, Ricardo A. Diagnóstico e modelagem cognitiva em ambientes de ensino inteligentes. Porto Alegre: CPGCC da UFRGS, 1996. Trabalho individual.

SILVEIRA, Ricardo Azambuja. Ambientes inteligentes distribuídos de aprendizagem. Porto Alegre: CPGCC da UFRGS, 1998.

TIJIBOY, Ana V., PEREIRA, Eliane A., WOICIECHOSKI, Lediane R. Interação com afeto: aprendizagem em ambientes virtuais de aprendizagem. Porto Alegre, RS: Renote - Revista Novas Tecnologias Na Educação, v. 7, n. 1, jul. 2009. Disponível em: <<http://seer.ufrgs.br/renote/article/view/13986/7878>>. Acesso em: 21 fev. 2013.

VALENTE José Armando. Informática na Educação: conformar ou transformar a escola. 2009.

VICCARI, Rosa Maria - Um tutor inteligente para a programação em lógica: idealização, projecto e desenvolvimento. Coimbra, 1989.

WEINBRENNER Stefan H, et al. *Supporting Cognitive Competence Development in Virtual Classroom*, in *10th International Conference on Advanced Learning Technologies*. IEEE 2010.

WOOLDRIDGE Michael. *An introduction to multiagent systems*. 2nd. Ed. Jonh Wiley & Sons LTd. 2009.

WOOLDRIDGE Michael. *Reasoning about rational agents*. In *Intelligent robots and autonomous agents*.2000.

ZAKRZEWSKA Danuta. *Cluster Analysis in Personalized E-Learning Systems*, in *Springer-Verlag Berlin Heidelberg*, 2009.

APÊNDICE A – Telas do sistema de configuração do agente

The screenshot shows the Moodle interface for course C0002. The top navigation bar includes the Moodle logo, the course path "Página inicial > Meus cursos > C0002", and a user profile section for "Professor" with a "Sair" (Logout) link and a language dropdown set to "Português - Brasil (pt_br)". A button labeled "Ativar edição" (Enable editing) is visible in the top right.

The main content area is titled "Programação" (Scheduling) and shows a list of activities under two units:

- Unidade 1**
 - Leitura_1
 - Atividade_1
- Unidade 2**
 - Leitura_2_Nível_Básico
 - Atividade_2_Nível_Básico
 - Atividade_3_Geral
 - Leitura_2_Nível_Médio
 - Atividade_2_Nível_Médio
 - Leitura_2_Nível_Avançado
 - Atividade_2_Nível_Avançado

The left sidebar contains a "Navegação" (Navigation) menu with links to "Página inicial", "Minha página inicial", "Páginas do site", "Meu perfil", "Meus cursos", and a detailed list for course C0002 including "Participantes", "Relatórios", "Geral", "Tópico 1", "Tópico 2", "Tópico 3", "Tópico 4", and "CCO001". Below this is a "Configurações" (Settings) section with links for "Administração do curso", "Ativar edição", "Editar configurações", and "Acompanhamento de Conclusão".

This screenshot shows the "Recursos" (Resources) and "Atividades" (Activities) configuration page for course C0002. The breadcrumb trail is "Página inicial > Meus cursos > C0002 > Recursos + Atividades".

The left sidebar is identical to the previous screenshot, showing the "Navegação" and "Configurações" menus.


The main content area is titled "Selecione o nível das atividades (Geral, Básico, Médio, Avançado):" (Select the level of activities (General, Basic, Medium, Advanced):). It contains two sections:

- Recursos:** A table with dropdown menus for selecting the level of each resource.

Nível	Recursos
Geral	Leitura_1
Básico	Leitura_2_Nível_Básico
Médio	Leitura_2_Nível_Médio
Avançado	Leitura_2_Nível_Avançado
Geral	Leitura_3_Geral
Básico	Leitura_4_Nível_Básico
Médio	Leitura_4_Nível_Médio
Avançado	Leitura_4_Nível_Avançado
- Atividades:** A table with dropdown menus for selecting the level of each activity.

Nível	Atividades
Geral	Atividade_1
Básico	Atividade_2_Nível_Básico
Geral	Atividade_3_Geral
Médio	Atividade_2_Nível_Médio
Avançado	Atividade_2_Nível_Avançado
Básico	Atividade_4_Nível_Básico
Médio	Atividade_4_Nível_Médio
Avançado	Atividade_4_Nível_Avançado

At the bottom of the page, there is a button labeled "Grava Perfil" (Save Profile).



Página inicial ► Meus cursos ► C0002 ► Recursos + Atividades

Navegação

- Página inicial
 - Minha página inicial
- Páginas do site
- Meu perfil
- Meus cursos
 - C0002**
 - Participantes
 - Relatórios
 - Geral
 - Tópico 1
 - Tópico 2
 - Tópico 3
 - Tópico 4
 - CCO001

Configurações

- Administração do curso
 - Ativar edição
 - Editar configurações
 - Acompanhamento de Conclusão
 - Usuários


Selecione um recurso e uma atividade para iniciar o estudo:

Recursos:

- ☒ Leitura_1
- ☐ Leitura_2_Nível_Básico
- ☐ Leitura_2_Nível_Médio
- ☐ Leitura_2_Nível_Avançado
- ☐ Leitura_3_Geral
- ☐ Leitura_4_Nível_Básico
- ☐ Leitura_4_Nível_Médio
- ☐ Leitura_4_Nível_Avançado

Atividades:

- ☒ Atividade_1
- ☐ Atividade_2_Nível_Básico
- ☐ Atividade_3_Geral
- ☐ Atividade_2_Nível_Médio
- ☐ Atividade_2_Nível_Avançado
- ☐ Atividade_4_Nível_Básico
- ☐ Atividade_4_Nível_Médio
- ☐ Atividade_4_Nível_Avançado



Você acessou como Professor - (Sair)

Português - Brasil (pt_br)

Página inicial ► Meus cursos ► C0002 ► Recursos + Atividades

Navegação

- Página inicial
 - Minha página inicial
- Páginas do site
- Meu perfil
- Meus cursos
 - C0002**
 - Participantes
 - Relatórios
 - Geral
 - Tópico 1
 - Tópico 2
 - Tópico 3
 - Tópico 4
 - CCO001

Configurações

- Administração do curso
 - Ativar edição
 - Editar configurações
 - Acompanhamento de Conclusão

Recurso e atividade iniciais

Recurso Inicial: Leitura_1
Atividade Inicial: Atividade_1

Atenção: Editar os recursos e atividades iniciais implica em apagar no banco de dados todos os pré-requisitos definidos até o momento.

Selecione um recurso ou uma atividade para informar seus pré-requisitos

Recursos e atividades:

-
- ☒ Leitura_2_Nível_Básico
- ☐ Leitura_2_Nível_Médio
- ☐ Leitura_2_Nível_Avançado
- ☐ Leitura_3_Geral
- ☐ Leitura_4_Nível_Básico
- ☐ Leitura_4_Nível_Médio
- ☐ Leitura_4_Nível_Avançado
- ☐ Atividade_2_Nível_Básico
- ☐ Atividade_3_Geral
- ☐ Atividade_2_Nível_Médio
- ☐ Atividade_2_Nível_Avançado
- ☐ Atividade_4_Nível_Básico
- ☐ Atividade_4_Nível_Médio

Editar

Atenção: Editar os recursos e atividades iniciais implica em apagar no banco de dados todos os pré-requisitos definidos até o momento.

Selecione um recurso ou uma atividade para informar seus pré-requisitos

Recursos e atividades:

Leitura_2_Nível ▾

Selecione os pré-requisitos

Recursos:

- ☒ Leitura_1
- ☐ Leitura_2_Nível_Médio
- ☐ Leitura_2_Nível_Avançado
- ☐ Leitura_3_Geral
- ☐ Leitura_4_Nível_Básico
- ☐ Leitura_4_Nível_Médio
- ☐ Leitura_4_Nível_Avançado

Atividades:

- ☒ Atividade_1
- ☐ Atividade_2_Nível_Básico
- ☐ Atividade_3_Geral
- ☐ Atividade_2_Nível_Médio
- ☐ Atividade_2_Nível_Avançado
- ☐ Atividade_4_Nível_Básico
- ☐ Atividade_4_Nível_Médio
- ☐ Atividade_4_Nível_Avançado

Recurso e atividade iniciais

Recurso Inicial: Leitura_1
Atividade Inicial: Atividade_1

Editar

Atenção: Editar os recursos e atividades iniciais implica em apagar no banco de dados todos os pré-requisitos definidos até o momento.

Selecione um recurso ou uma atividade para informar seus pré-requisitos

Recursos/atividade:

Leitura_2_Nível ▾

Pré-requisitos selecionados

Recursos:
Leitura_1

Atividades:
Atividade_1

Indicar pré-requisitos de outros recursos e atividades

Continuar

Finalizar

Recurso e atividade iniciais

Recurso Inicial: Leitura_1

Atividade Inicial: Atividade_1

Árvore de Pré-requisitos da turma

